

## 第四章

# 順序邏輯

- 4-1 正反器
- 4-2 非同步計數器
- 4-3 同步計數器
- 4-4 利用 IC 模組製作計數器
- 4-5 微分電路
- 4-6 防彈跳電路
- 4-7 動動腦 順序邏輯練習

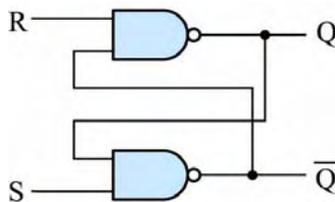
組合邏輯電路的輸出各種狀態僅與其電路當時的輸入有關，而順序邏輯電路則是目前輸入、輸出狀態皆影響下一個輸入輸出狀態。它比組合邏輯電路多了一個含有記憶元件的回授網路，使它上一步的輸出結果會影響到下一步，順序邏輯電路中，除了邏輯閘電路外，尚有一個很重要的元件，就是正反器 (flip-flop，簡稱 FF)，它是個雙穩態多諧振盪器，輸出具有兩種穩定狀態。它有兩個輸出端，分別輸出相反的狀態，即當其一端輸出高態時，另一端為其反相輸出成為低態。它是個具有記憶裝置的電路，在一穩定狀態輸出下，若沒有觸發信號使它改變輸出狀態，它會維持此一固定狀態輸出。以下各節將分別介紹順序邏輯之基本及應用電路。

## 4-1 正反器

### 4-1-1 RS 閘(Latch)

RS 閘可分成兩種結構，一種是由 NAND 閘組成，另一種則是由 NOR 閘組成。電路有兩輸入端，分別命名 S 及 R，S 代表設置端(Set)，R 代表重置端(Reset)。輸出端有兩個，Q 代表其輸出端， $\bar{Q}$  表示為反相輸出端。其電路結構及真值表如下：

#### I. NAND 閘組成電路及真值表



● 圖 4-1-1 NAND 閘組成 RS 閘電路

● 表 4-1-1 NAND 閘 RS 閘真值表一

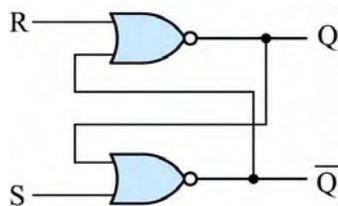
S	R	$Q_{n+1}$
0	0	不合邏輯
0	1	0
1	0	1
1	1	$Q_n$

● 表 4-1-2 NAND 閘 RS 閘真值表二

S	R	$Q_n$	$Q_{n+1}$
0	0	0	不合邏輯
0	0	1	不合邏輯
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

由電路圖 4-1-1 可知，兩 NAND 的輸出皆回授至另一個 NAND 閘的輸入，此種結構稱 RS 閘。其中以 S 及 R 代表兩輸入端，兩輸出端分別為輸出狀態 Q 及其反相輸出  $\bar{Q}$ 。真值表一中， $Q_n$  代表現在的輸出狀態， $Q_{n+1}$  代表下一個狀態的輸出。當 S 及 R 兩輸入端皆為 0 時，因為 NAND 閘只要有一輸入端為 0，其輸出即為 1。故在此種情況下，Q 與  $\bar{Q}$  皆變成 1，而任一變數及該變數的反相不可能同時為 1，所以此輸入狀態不合邏輯，不可在電路中出現，在使用上要避免用到此種情況。當 S=0，R=1 時， $Q_{n+1}=0$ ，代表下一個狀態為 0。當 S=1，R=0 時， $Q_{n+1}=1$ ，代表下一個狀態為 1。當 S=1，R=1 時， $Q_{n+1}=Q_n$ ，代表下一個狀態為與現在的狀態一樣，維持不變。因此，表 4-1-1 真值表的寫法也可寫成如表 4-1-2 真值表二的狀態。

## 2. NOR 閘組成



● 圖 4-1-2 NOR 閘組成 RS 閘電路

● 表 4-1-3 NOR 閘 RS 閘真值表一

S	R	$Q_{n+1}$
0	0	$Q_n$
0	1	0
1	0	1
1	1	不合邏輯

● 表 4-1-4 NOR 閘 RS 閘真值表二

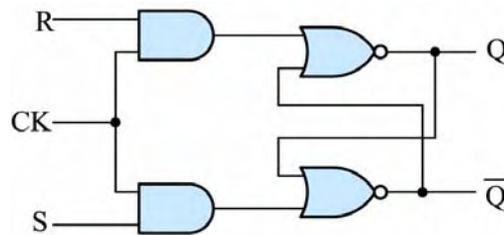
S	R	$Q_n$	$Q_{n+1}$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	不合邏輯
1	1	1	不合邏輯

由電路圖 4-1-2 可知，兩 NOR 的輸出皆回授至另一個 NOR 閘的輸入，此種結構稱 RS 閘。其中以 S 及 R 代表兩輸入端，兩輸出端分別為輸出狀態 Q 及其反相輸出  $\bar{Q}$ 。真值表一中， $Q_n$  代表現在的輸出狀態， $Q_{n+1}$  代表下一個狀態的輸出。當 S 及 R 兩輸入端皆為 1 時，因為 NOR 閘只要有一輸入端為 1，其輸出即為 0。故在此種情況下，Q 與  $\bar{Q}$  將變成 0，而任一變數及該變數的反相不可能同時為 0，所以此輸入狀態不合邏輯，不可在電路中出現，在使用上要避免用到此種情況。當 S=0，R=1 時， $Q_{n+1}=0$ ，代表下一個狀態為 0。當 S=1，R=0 時， $Q_{n+1}=1$ ，代表下一個狀態為 1。當 S=0，R=0 時， $Q_{n+1}=Q_n$ ，代表下一個狀態為與現在的狀態一樣，維持不變。因此，表 4-1-3 真值表一的寫法也可寫成如表 4-1-4 真值表二的狀態。

### 4-1-2 RS 正反器

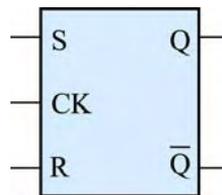
在 RS 閘中，當 R 與 S 狀態一有改變時，輸出即會有相對應的變化，而很難明確定出下一個狀態是何時。因此，若在輸入端部份配合邏輯閘以及時脈 (Clock) 信號，便能以一個時脈週期當成一個狀態的時間，便有了 RS 正反器。

#### 1. 電路圖



● 圖 4-1-3 NOR 閘組成 RS 閘電路

#### 2. 符號



● 圖 4-1-4

#### 3. 真值表

● 表 4-1-5 RS 正反器真值表一

S	R	CK	$Q_{n+1}$
$\phi$	$\phi$	0	$Q_n$
0	0	1	$Q_n$
0	1	1	0
1	0	1	1
1	1	1	不合邏輯

表 4-1-6 RS 正反器真值表二

S	R	$Q_n$	CK	$Q_{n+1}$
$\phi$	$\phi$	0	0	0
$\phi$	$\phi$	1	0	1
0	0	0	1	0
0	0	1	1	1
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	不合邏輯
1	1	1	1	不合邏輯

由電路圖 4-1-3 可知，當 CK=0 時，AND 閘輸出為 0，此時 NOR 閘 RS 閘的輸入為 00，代表  $Q_n$  狀態維持不變。只有當 CK=1 時，RS 閘才會隨著 R 與 S 的輸入改變狀態。此表示每次 CK=1 時，正反器狀態改變一次。真值表中， $\phi$  (don't care) 代表可為 0 或 1，即表示當 CK=0 時，不管 R 與 S 狀態是 0 或 1，輸出皆不受影響，維持不變。當 CK=1 時，則輸出有如一般之 RS 閘。表 4-1-5 真值表一的寫法也可寫成如表 4-1-6 真值表二的狀態。

#### 4. 特性方程式(Characteristic equation)

由表 4-1-6 真值表二可得輸出端  $Q_{n+1}$  與輸入 R,S, $Q_n$  間布林函數的關係如下：

$$Q_{n+1} = \overline{R}SQ_n + \overline{R}\overline{S}\overline{Q}_n + \overline{R}\overline{S}Q_n$$

經由化簡，可知輸出的特性方程式如下：

$$Q_{n+1} = S + \overline{R}Q_n$$

$$SR = 0$$

(其中  $SR=0$  表示 S 與 R 不可同時為 1，因為會產生不合邏輯的現象)

#### 5. 激勵表：

由真值表可知輸入對輸出的狀態變化，但若已知輸出的狀態變化情形，想要推論輸入是什麼狀態，則需激勵表。如表 4-1-8 所示為 RS 正反器之真值表，由真值表可知當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 0 變成 0 時，這時的輸入 S 與 R 有兩

種可能，一是  $SR=00$ ，一是  $SR=01$ 。由此可知，只要  $S=0$ ，不管  $R$  是 0 或 1，當  $Q_n$  為 0 時，它的下一狀態  $Q_{n+1}$  都會維持為 0。當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 0 變成 1 時，這時的輸入  $S$  與  $R$  只有一種可能，就是  $SR=10$ 。由此可知，只要  $S=1$ ， $R$  是 0，當  $Q_n$  為 0 時，它的下一狀態  $Q_{n+1}$  都會轉態為 1。當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 1 變成 0 時，這時的輸入  $S$  與  $R$  只有一種可能，就是  $SR=01$ 。由此可知，只要  $S=0$ ， $R$  是 1，當  $Q_n$  為 1 時，它的下一狀態  $Q_{n+1}$  都會轉態為 0。當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 1 變成 1 時，這時的輸入  $S$  與  $R$  有兩種可能，一是  $SR=00$ ，一是  $SR=10$ 。由此可知，只要  $R=0$ ，不管  $S$  是 0 或 1，當  $Q_n$  為 1 時，它的下一狀態  $Q_{n+1}$  都會維持為 1。由此便可求得 RS 正反器的激勵表。如表所示， $Q_n \rightarrow Q_{n+1}$  表示輸出端的狀態改變， $0 \rightarrow 0$  表示輸出端  $Q_n = 0$  變成下一狀態時  $Q_{n+1} = 0$ ，以此類推。 $\phi$  (don't care) 表示其值可以是 0 或 1 而不影響結果。

表 4-1-7 RS 正反器激勵表

$Q_n \rightarrow Q_{n+1}$	S	R
$0 \rightarrow 0$	0	$\phi$
$0 \rightarrow 1$	1	0
$1 \rightarrow 0$	0	1
$1 \rightarrow 1$	$\phi$	0

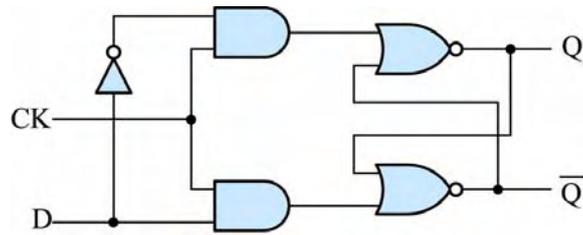
表 4-1-8 RS 正反器之真值表三

S	R	$Q_n$	$Q_{n+1}$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	不合邏輯
1	1	1	不合邏輯

### 4-1-3 D 型正反器

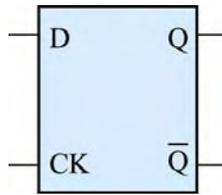
將 RS 正反器之兩輸入端經由一個反閘(NOT 閘)連接而變成一個輸入端 D，即成為 D 型正反器。其特性為當時脈  $CK=1$  時，輸入若  $D=0$ ，則輸出  $Q_n = 0$ ，輸入若  $D=1$ ，則輸出  $Q_n = 1$ 。

1. 電路圖



● 圖 4-1-5 D 型正反器電路圖

2. 符號



● 圖 4-1-6

3. 真值表

● 表 4-1-9 D 型正反器之真值表一

D	CK	$Q_{n+1}$
$\phi$	0	$Q_n$
0	1	0
1	1	1

● 表 4-1-10 D 型正反器之真值表二

D	$Q_n$	CK	$Q_{n+1}$
$\phi$	0	0	0
$\phi$	1	0	1
0	0	1	0
0	1	1	0
1	0	1	1
1	1	1	1

4. 特性方程式

由真值表知，其下一狀態之輸出  $Q_{n+1}$  與輸入 D 及現在狀態  $Q_n$  之布林函數為：

$$Q_{n+1} = D$$

### 5. 激勵表

由真值表可知當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 0 變成 0 時，D 一定是 0，當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 0 變成 1 時，D 一定是 1，當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 1 變成 0 時，D 一定是 0，當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 1 變成 1 時，D 一定是 1。由此可得如表 4-1-11 D 型正反器激勵表。

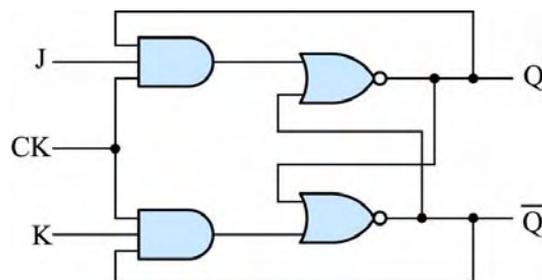
● 表 4-1-11 D 型正反器激勵表

$Q_n \rightarrow Q_{n+1}$	D
0→0	0
0→1	1
1→0	0
1→1	1

### 4-1-4 JK 正反器

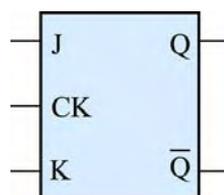
將 RS 正反器之兩輸入端經由兩個三輸入及閘(AND 閘)連接即可變成如圖之 JK 正反器。其特性比 RS 正反器多了一個恆變(輸出為輸入之反相)的功能。為當時脈  $CK=1$  時，輸入若  $JK=00$ ，則輸出維持不變  $Q_{n+1}=Q_n$ ，輸入若  $JK=01$ ，則輸出  $Q_{n+1}=0$ ，輸入若  $JK=10$ ，則輸出  $Q_{n+1}=1$ ，輸入若  $JK=11$ ，則輸出變成恆變  $Q_{n+1}=\bar{Q}_n$ 。

#### 1. 電路圖



● 圖 4-1-7 JK 正反器電路圖

#### 2. 符號



● 圖 4-1-8

## 3. 真值表

表 4-1-12 JK 正反器真值表一

J	K	CK	$Q_{n+1}$
$\phi$	$\phi$	0	$Q_n$
0	0	1	$Q_n$
0	1	1	0
1	0	1	1
1	1	1	$\overline{Q_n}$

表 4-1-13 JK 正反器真值表二

J	K	$Q_n$	CK	$Q_{n+1}$
$\phi$	$\phi$	0	0	0
$\phi$	$\phi$	1	0	1
0	0	0	1	0
0	0	1	1	1
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	0

## 4. 特性方程式

由表 4-1-13 真值表二可知下一狀態之輸出  $Q_{n+1}$  與輸入 J, K 及現在狀態  $Q_n$  之布林函數為：

$$Q_{n+1} = \overline{J} \overline{K} Q_n + J \overline{K} Q_n + J \overline{K} \overline{Q_n} + JK \overline{Q_n}$$

經由化簡可得特性方程式：

$$Q_{n+1} = JQ_n + \overline{K}Q_n$$

## 5. 激勵表

如表 4-1-13 所示為 JK 正反器之真值表，由真值表可知當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 0 變成 0 時，這時的輸入 J 與 K 有兩種可能，一是 JK=00，一是 JK=01。由此可知，只要 J=0，不管 K 是 0 或 1，當  $Q_n$  為 0 時，它的下一狀態  $Q_{n+1}$  都會維持為 0。當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 0 變成 1 時，這時的輸入 J 與 K 有兩種可能，一是 JK=10，一是 JK=11。由此可知，只要 J=1，不管 K 是 0 或 1，當  $Q_n$

為 0 時，它的下一狀態  $Q_{n+1}$  都會轉態為 1。當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 1 變成 0 時，這時的輸入 J 與 K 有兩種可能，一是  $JK=01$ ，一是  $JK=11$ 。由此可知，只要  $K=1$ ，不管 J 是 0 或 1，當  $Q_n$  為 1 時，它的下一狀態  $Q_{n+1}$  都會轉態為 0。當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 1 變成 1 時，這時的輸入 J 與 K 有兩種可能，一是  $JK=00$ ，一是  $JK=10$ 。由此可知，只要  $K=0$ ，不管 J 是 0 或 1，當  $Q_n$  為 1 時，它的下一狀態  $Q_{n+1}$  都會維持為 1。由此便可求得 JK 正反器的激勵表。

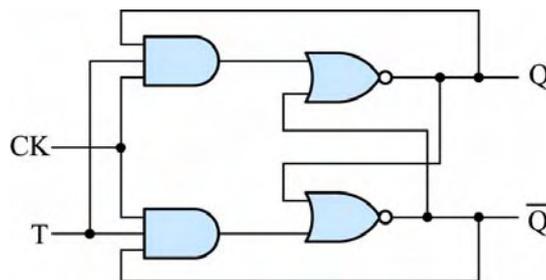
● 表 4-1-13 JK 正反器激勵表

$Q_n \rightarrow Q_{n+1}$	J	K
0→0	0	$\phi$
0→1	1	$\phi$
1→0	$\phi$	1
1→1	$\phi$	0

#### 4-1-5 T 型正反器

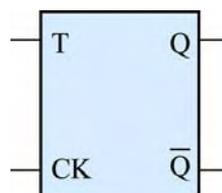
將 JK 正反器之兩輸入端 J 與 K 連接在一起變成單一輸入端，稱為 T 型正反器，此 T 型正反器只有兩種狀態，當  $T=0$  時，輸出維持不變，下一個輸出狀態與現在狀態相同， $Q_{n+1}=Q_n$ 。當  $T=1$  時，輸出為恆變，下一個輸出狀態與現在狀態相反， $Q_{n+1}=\bar{Q}_n$ 。

##### 1. 電路圖



● 圖 4-1-9 T 型正反器電路圖

##### 2. 符號



● 圖 4-1-10

## 3. 真值表

表 4-1-15 T 型正反器真值表一

T	CK	$Q_{n+1}$
$\phi$	0	$Q_n$
0	1	$Q_n$
1	1	$\overline{Q_n}$

表 4-1-16 T 型正反器真值表二

T	$Q_n$	CK	$Q_{n+1}$
$\phi$	0	0	0
$\phi$	1	0	1
0	0	1	0
0	1	1	1
1	0	1	1
1	1	1	0

## 4. 特性方程式

由表 4-1-16 真值表二可知下一狀態之輸出  $Q_{n+1}$  與輸入 T 及現在狀態  $Q_n$  之布林函數為：

$$Q_{n+1} = \overline{T}Q_n + T\overline{Q}_n$$

$$Q_{n+1} = T \oplus Q_n$$

## 5. 激勵表

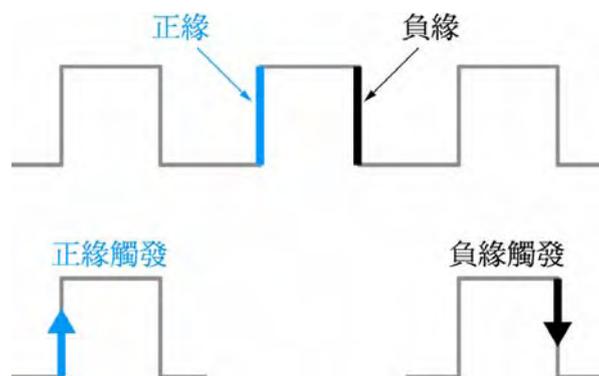
由表 4-1-16 真值表二可知當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 0 變成 0 時，T 一定是 0，當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 0 變成 1 時，T 一定是 1，當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 1 變成 0 時，T 一定是 1，當  $Q_n$  變成  $Q_{n+1}$  的狀態是由 1 變成 1 時，T 一定是 0。由此可得如表 4-1-17 激勵表。

表 4-1-17 T 型正反器激勵表

$Q_n \rightarrow Q_{n+1}$	T
0→0	0
0→1	1
1→0	1
1→1	0

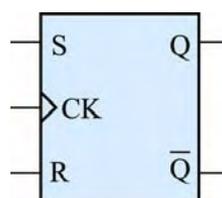
### 4-1-6 邊緣觸發之 RS 正反器

由上述真值表可知，當時序信號為 1(高態)時，正反器電路輸出狀態會隨輸入改變，但當時序信號的高態若維持很久，則很難得知正反器輸出狀態改變幾次。在順序邏輯中，基本上希望時序信號產生一次高態時，正反器輸出只改變一次，因此有了邊緣觸發(edge triggering)的正反器，即正反器在時序信號狀態改變時才觸發，輸出才隨著改變。當時序信號由 0(低態)變成 1(高態)時，正反器輸出狀態改變者稱正緣觸發(positive edge triggering)，當時序信號由 1 變成 0 時，正反器輸出狀態改變者稱負緣觸發(negative edge triggering)。



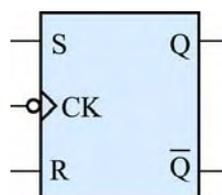
● 圖 4-1-11

- (1) 正緣觸發之 RS 正反器符號為在時脈輸入端(CK)處多一個三角形，負緣觸發則是又多了一個圓圈，用以代表反相。
- (2) 正緣觸發之 RS 正反器符號

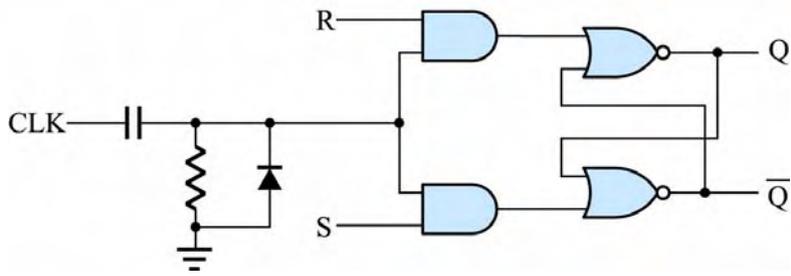


● 圖 4-1-12

- (3) 負緣觸發之 RS 正反器符號



● 圖 4-1-13

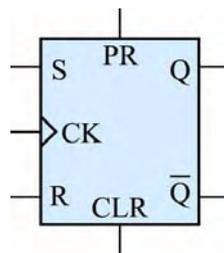


● 圖 4-1-14 正緣觸發之 RS 正反器電路圖

如圖 4-1-14 所示，在 CLK 時脈輸入 RS 正反器前，事先經過微分電路，使得輸入時脈僅在瞬間維持為 1(高態)，正反器在該瞬間轉態，即為正緣觸發之 RS 正反器。將輸入時脈接上反相器再輸入微分電路則可成為負緣觸發之 RS 正反器，上述介紹之 D 型正反器、JK 正反器、T 型正反器也都可變成邊緣觸發之正反器。

#### 4-1-7 可預設、清除之正反器

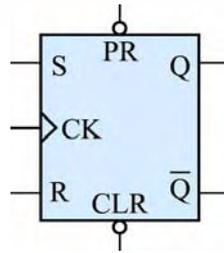
由於順序邏輯電路，常需用到預設(Preset)與清除(Clear)功能，所以一般正反器通常都有此功能。在所有輸入控制端中，此兩者具有最高優先權。其分成兩種，圖 4-1-15 表高態動作之正反器，動作如真值表所示，當 CLR=1 時， $Q_n=0$ ，當 PR=1， $Q_n=1$ ，若兩者同時為 1 時，則視 PR 與 CLR 何者優先，若 PR 優先則為 1，CLR 優先則為 0。圖 4-1-16 表低態動作之正反器，動作如真值表所示，當 CLR=0 時， $Q_n=0$ ，當 PR=0， $Q_n=1$ ，若兩者同時為 0，則視 PR 與 CLR 何者優先，若 PR 優先則為 1，CLR 優先則為 0。



● 圖 4-1-15 可預設、清除之正反器符號

■ 表 4-1-18 可預設、清除之正反器真值表

PR	CLR	$Q_{n+1}$
0	0	$Q_n$
0	1	0
1	0	0
1	1	依優先次序決定



● 圖 4-1-16 可預設、清除之正反器符號

■ 表 4-1-19 可預設、清除之正反器真值表

PR	CLR	$Q_{n+1}$
0	0	依優先次序決定
0	1	1
1	0	0
1	1	$Q_n$

## 4-2 非同步計數器

正反器可用來做計數器(counter)電路，一個正反器可產生兩種輸出狀態， $n$ 個正反器產生  $2^n$  種輸出狀態。利用這種依序輸出不同狀態的特性可用來當計數器使用。在這計數器中，若是所有正反器的時脈輸入端是接在一起，則表示這些正反器是同時做輸出轉態動作，稱之為同步計數器(synchronous counter)。反之若是這些正反器的時脈輸入端未接在一起，而是由一個正反器的輸出端接到另一個正反器的時脈輸入端，表示前面正反器動作完成後，後面正反器才有反應，並未同時動作，稱之為非同步計數器(asynchronous counter)，因為是前一個正反器推動後一個正反器，有如漣漪般散開，又稱之為漣波(ripple)計數器。

每個正反器從時脈輸入觸發到輸出產生變化需要一段時間，稱為傳遞延遲時間( $t_p$ , propagation delay time)或簡稱延遲時間。因為非同步計數器的正反器是由前一個正反器的輸出來觸發，因此輸出信號所需的延遲時間為正反器延遲時間的累加，所以速度較慢，但硬體線路簡單，若有  $n$  個正反器，其傳遞延遲時間最長為  $n t_p$ ，其最大工作頻率  $f_{max} = 1/(n t_p)$ 。而同步計數器的每一個正反器是由同一個時脈信號同時觸發，因此輸出信號所需的延遲時間為一個正反器的延遲時間，所以傳送速度較快，但相對的硬體線路較為複雜。計數器的應用很廣泛，如頻率、週期及時間方面測量。以下將先介紹各種非同步計數器。

### 4-2-1 JK 正反器之計數器

由表 4-2-1 JK 正反器之真值表可知，若  $J=1$ ， $K=1$ ，且下一個時脈來時，輸出的狀態會與現在的狀態相反  $Q_{n+1}=\overline{Q}_n$ 。如圖 4-2-1 所示，為 JK 正反器之 J、K 輸入端接高態(H，即邏輯 1)，當輸入時脈 CLK 之波形輸入，則將產生相對輸出波形如  $Q_n$ 。若使用負緣觸發，每一個時脈的負緣來時，促使正反器轉態一次，而每兩個時脈負緣將使正反器輸出一個高態及一個低態，即一個脈波。由此可知，JK 正反器此時變成一個除頻電路，輸出頻率為輸入時脈頻率的 1/2。

表 4-2-1 JK 正反器真值表

J	K	CK	$Q_{n+1}$
$\phi$	$\phi$	0	$Q_n$
0	0	1	$Q_n$
0	1	1	0
1	0	1	1
1	1	1	$\overline{Q}_n$

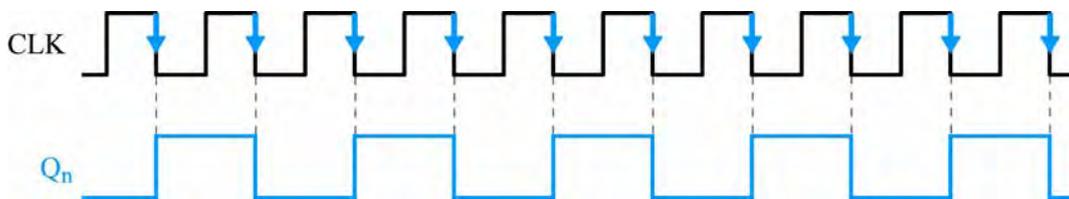


圖 4-2-1 JK 正反器之  $J=K=1$  時之波形圖

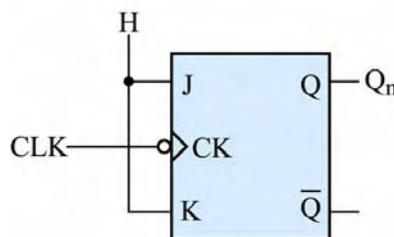


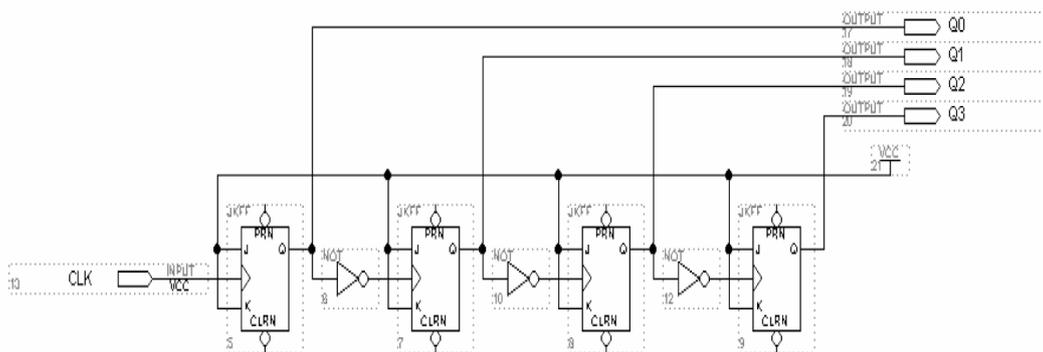
圖 4-2-2 JK 正反器之  $J=K=1$  時之電路圖

利用 1 個正反器電路可產生兩種輸出狀態(0,1)，稱之為 Mode 2 (或 MOD 2) 電路，把它當成除頻電路則其功能可以除 2。若是利用兩個正反器則可產生四種輸出狀態( $Q_1Q_0$  可為 00,01,10,11)，稱之為 Mode 4 (或 MOD 4) 電路，把它當成除頻電路則其功能可以除 4。依此類推，利用  $n$  個正反器則可產生  $2^n$  種輸出狀態，稱之為 Mode  $2^n$  (或 MOD  $2^n$ ) 電路，把它當成除頻電路則其功能可以除  $2^n$ 。除頻

電路亦可用來當計數器使用，一個正反器的除頻電路相當於一位元之計數器，有兩種狀態，可產生 0,1 之計數；二個正反器的除頻電路相當於二位元之計數器，有四種狀態，可產生 00,01,10,11 之輸出，即 0,1,2,3 之二進制計數器； $n$  個正反器的除頻電路相當於  $n$  位元之計數器有  $2^n$  種狀態，可產生 0,1,2,3,..., $2^n-1$  之二進制計數器。而且，並非只能產生  $2^n$  之二進制計數器，若是使用  $n$  個正反器，只要配合正反器清除(CLR)、預置(PRS)及適當邏輯電路，就可產生任何  $2^n$  種狀態以下的計數器。

### 一、MOD16 上數計數器

使用 4 個正反器可產生 16 種狀態，可用來當 0~15 的二進制計數器。當產生的計數器是由 0,1,2,3,... 由小而大依序往上計數者稱為上數計數器(up counter)，製作上數計數器之正反器需用負緣觸發，如此才能剛好在前面正反器波形一轉態時，觸發後面正反器做上數之動作。反之若是由大而小依序往下計數者，稱為下數計數器(down counter)，下數計數器則需使用正緣觸發之正反器。以下將介紹 MOD16 上數計數器的作法。

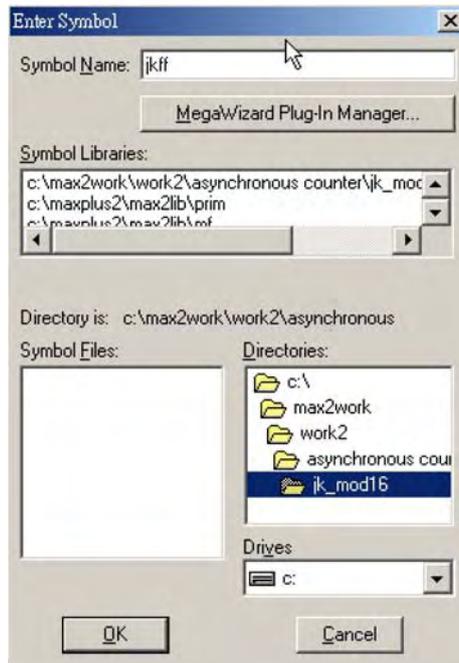


● 圖 4-2-3 MOD 16 上數計數器電路圖

如圖 4-2-3 所示經由步驟：[畫電路圖](#)→[存檔](#)→[設為工作專案](#)→[指定 CPLD 晶片](#)→[編譯](#)→[軟體模擬](#)→[規劃腳位](#)→[編譯](#)→[下載燒錄](#)(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

**製作 MOD16 上數計數器步驟如下：**

1. 開啟新圖形編輯檔，繪製邏輯電路圖，如圖 4-2-3 所示，利用 [Symbol](#)→[Enter Symbol](#) 叫出 [Enter Symbol](#) 視窗，輸入 jkff 以叫出 JK 正反器(JK flip-flop)。因為此為正緣觸發，要做上數計數器需用負緣觸發，所以時脈輸入端前面需再接反閘(NOT)，依照前述方法畫出電路圖。



● 圖 4-2-4 元件取用視窗

2. 存檔，取檔名為 mod 16up.gdf
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

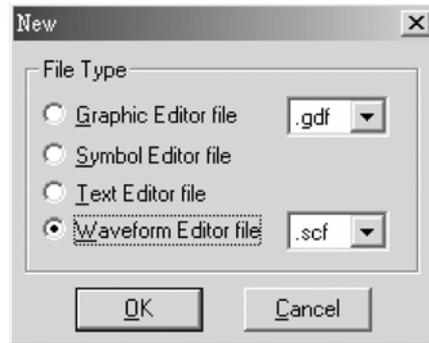
如圖 4-2-5 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



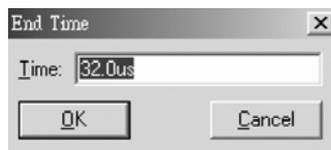
● 圖 4-2-5 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $32 \mu s$ ，設定格線間距(Options→Grid Size)設定  $1 \mu s$ ，顯示在視窗中適當大小格線(View→Fit in Window)。

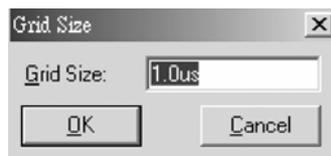
(因為在此計數器輸出有 4 個，可計數 0~15，總共有  $2^4=16$  種變化，當我們設 Grid= $1 \mu s$  時，一個時脈信號需要高態及低態各  $1 \mu s$ ，共需  $2 \mu s$ ，故計數 0~15 需 16 個時脈，總共需  $2 \mu s \times 16 = 32 \mu s$  才完成模擬。)



● 圖 4-2-6 開啟新檔視窗

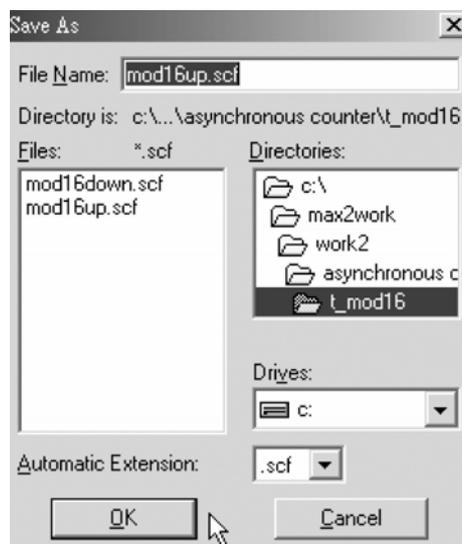


● 圖 4-2-7 模擬結束時間設定視窗

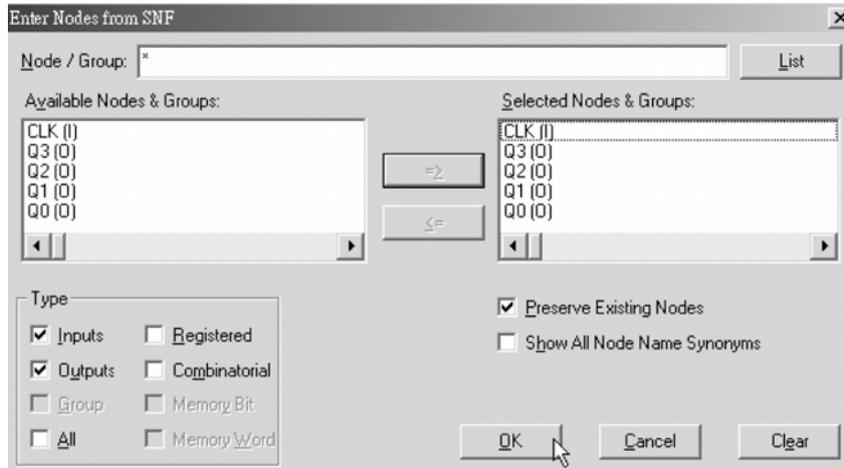


● 圖 4-2-8 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 mod 16up.scf，輸入節點(Node→Enter Nodes from SNF，按 List 及  $\Rightarrow$ ，OK)，編輯輸入信號，輸入端有五支腳，總共有三十二種組合，剛好  $32 \mu s$  可模擬完成，可用計數時脈完成輸入信號編輯。

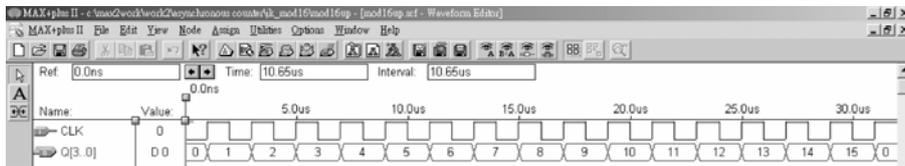


● 圖 4-2-9 儲存檔案視窗

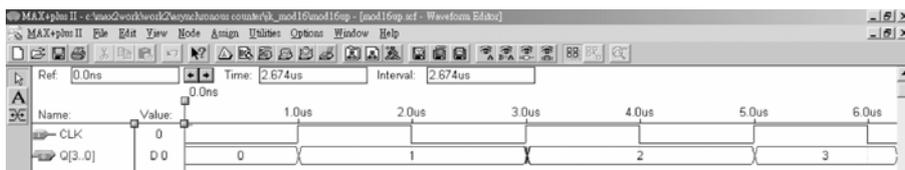


● 圖 4-2-10 輸出入節點選擇視窗

- 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-2-14 所示,可點選 Q[3..0],按滑鼠右鍵選擇 Ungroup,可得各輸出端波形圖。模擬沒有錯誤及警告,所得波形模擬結果符合 MOD16 上數計數器,代表我們製作的電路是正確可用的。

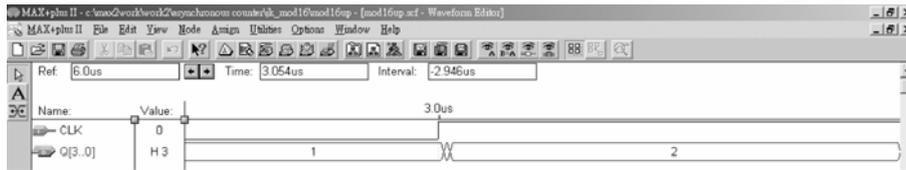


● 圖 4-2-11 模擬結果視窗

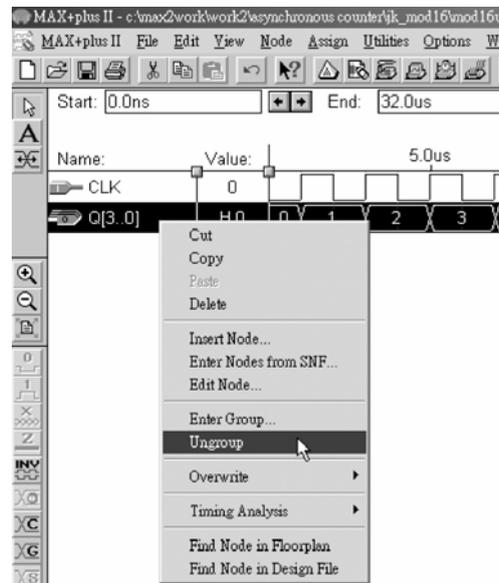


● 圖 4-2-12 模擬結果視窗

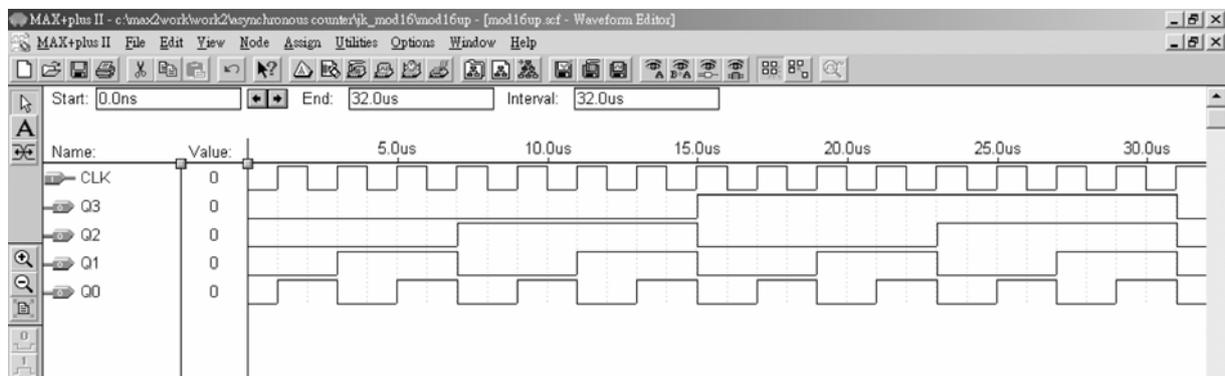
另外補充一點：利用 調整大小,可看出在有很多時間點有出現漣波未定態(如 3  $\mu$ s 時的上下振盪),這是因為非同步計數器並非同時由同一個時脈輸入信號觸發,因此其會有不同正反器產生的延遲觸發間距的缺點,這種現象在同步計數器則不會出現。



● 圖 4-2-13 模擬結果視窗



● 圖 4-2-14 模擬波形設定—解除群組化



● 圖 4-2-15 模擬結果視窗

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

● 表 4-2-2 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

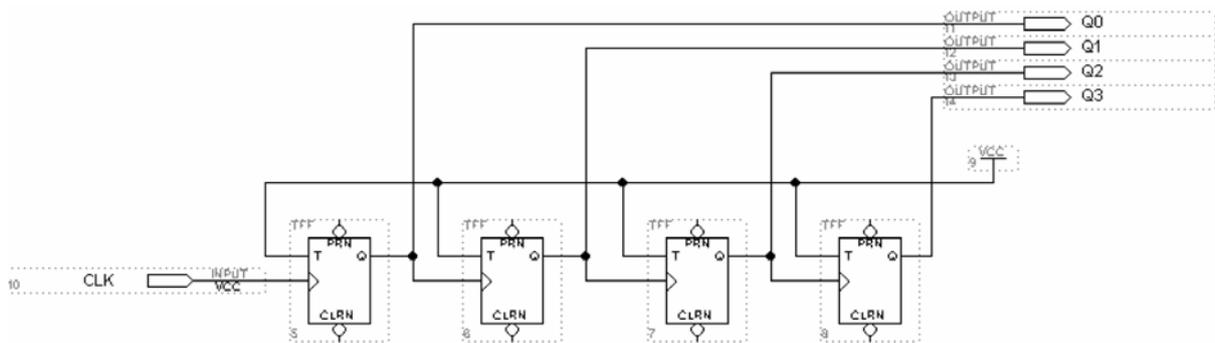
腳位對應關係	輸入	輸出			
名稱	CLK	Q0	Q1	Q2	Q3

CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76
實驗器模組對應腳位	CLK1	DG0	DG1	DG2	DG3

實驗器可選擇使用 CLK<sub>1</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR<sub>1</sub> 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3 個綠色 LED 接腳當輸出端 Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>。

## 二、MOD16 下數計數器

使用 4 個正反器可產生 16 種狀態，可用來當 0~15 的二進制計數器。當產生的計數器是由 15,14,13,...,0 由大而小依序往下計數者稱為 MOD16 下數計數器(down counter)，下數計數器需使用正緣觸發之正反器。以下將介紹 MOD16 下數計數器的作法。

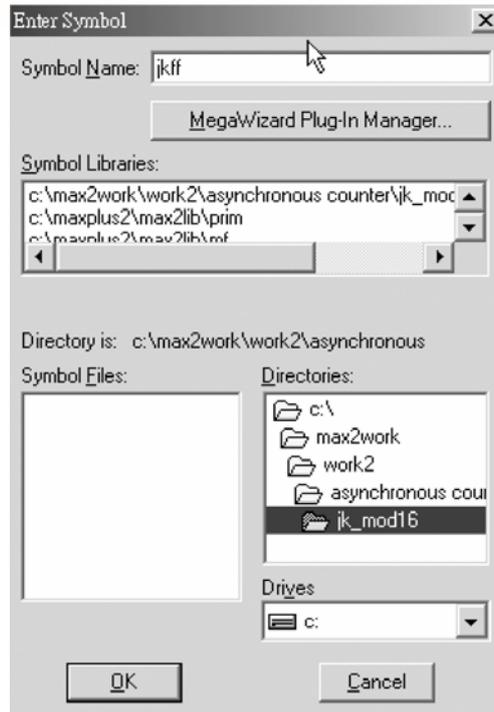


● 圖 4-2-16 MOD16 下數計數器電路圖

如圖 4-2-16 所示經由步驟：**畫電路圖**→**存檔**→**設為工作專案**→**指定 CPLD 晶片**→**編譯**→**軟體模擬**→**規劃腳位**→**編譯**→**下載燒錄**(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

### 製作 MOD16 下數計數器步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖，如圖 4-2-17 所示，利用 Symbol→ Enter Symbol 叫出 Enter Symbol 視窗，輸入 jkff 以叫出 JK 正反器(JK flip-flop)。此為正緣觸發，要做下數計數器需用正緣觸發，所以時脈輸入端前面無需再接反閘(NOT)，此與上數計數器不同，依照前述方法畫出電路圖。



● 圖 4-2-17 元件取用視窗

2. 存檔，取檔名為 mod 16 down.gdf。
3. 設為指定工作專案 [File→Project→Set Project to Current File](#)，指定 CPLD 晶片 [Assign→Device](#)，並編譯 [MAX+PLUS II → Compiler](#)。

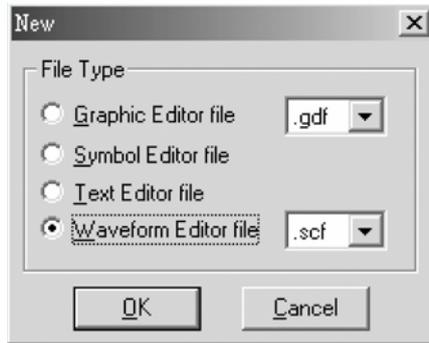
如圖所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



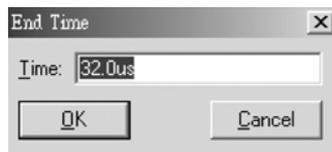
● 圖 4-2-18 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間([File→End Time](#))設定  $32 \mu s$ ，設定格線間距([Options → Grid Size](#))設定  $1 \mu s$ ，顯示在視窗中適當大小格線 ([View→Fit in Window](#))。

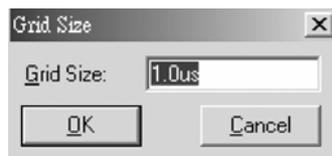
(因為在此計數器輸出有 4 個，可計數 0~15，總共有  $2^4=16$  種變化，當我們設  $Grid=1\mu s$  時，一個時脈信號需要高態及低態各  $1 \mu s$ ，共需  $2 \mu s$ ，故計數 0~15 需 16 個時脈，總共需  $2 \mu s \times 16 = 32 \mu s$  才完成模擬。)



● 圖 4-2-19 開啟新檔視窗

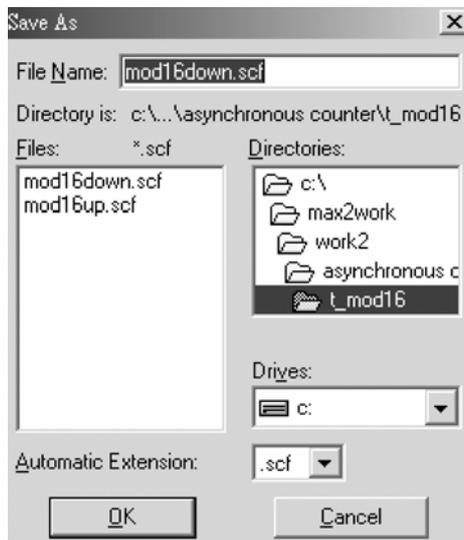


● 圖 4-2-20 模擬結束時間設定視窗

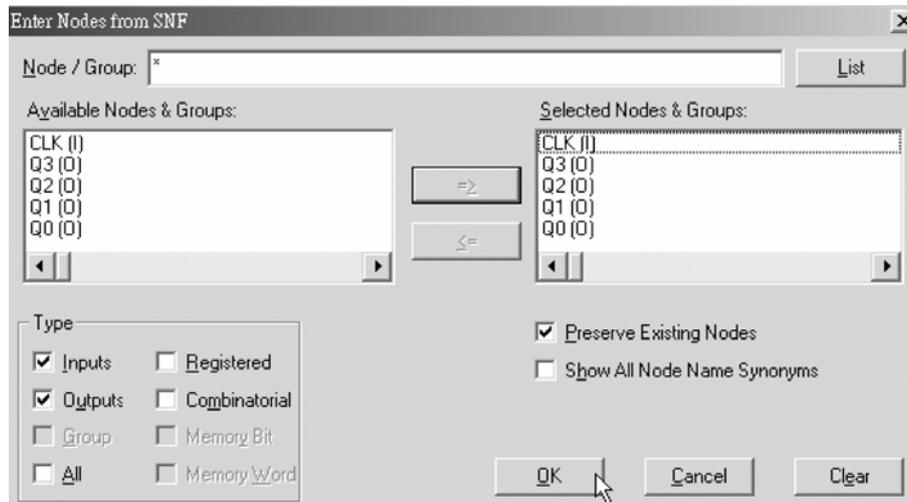


● 圖 4-2-21 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 mod 16 down.scf，輸入節點(Node → Enter Nodes from SNF，按 List 及 ⇒，OK)，編輯輸入信號，輸入端有五支腳，總共有三十二種組合，剛好 32 μs 可模擬完成，可用計數時脈完成輸入信號編輯。

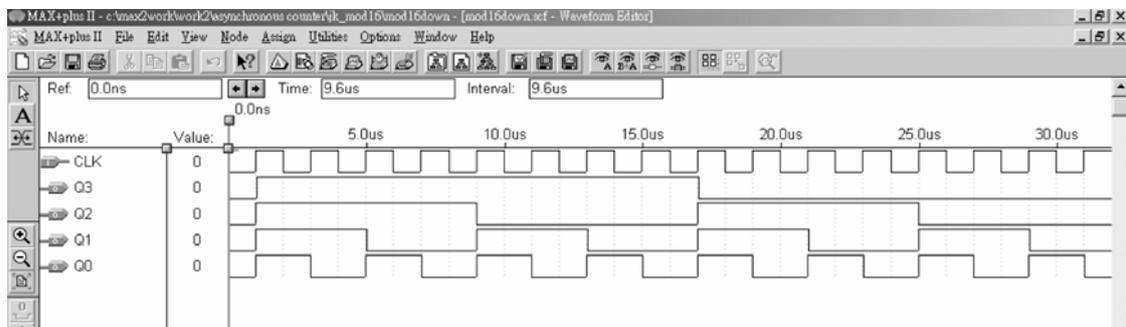


● 圖 4-2-22 儲存檔案視窗



● 圖 4-2-23 輸出入節點選擇視窗

6. 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-2-24 所示。模擬沒有錯誤及警告，所得波形模擬結果符合 MOD16 下數計數器，代表我們製作的電路是正確可用的。



● 圖 4-2-24 模擬結果視窗

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

● 表 4-2-3 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出			
名稱	CLK	Q0	Q1	Q2	Q3
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76
實驗器模組對應腳位	CLK1	DG0	DG1	DG2	DG3

實驗器可選擇使用 CLK<sub>1</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR<sub>1</sub> 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3 個綠色 LED 接腳當輸出端 Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>。

### 4-2-2 T 型正反器之計數器

由 JK 正反器所做的計數器，主要是用到當 J=1，K=1，在下一個時脈來時，輸出的狀態會與現在的狀態相反  $Q_{n+1} = \overline{Q}_n$ 。此種功能 T 型正反器亦有具備，當 T 型正反器之輸入信號 T=1 時，在下一個時脈來時，輸出的狀態亦會與現在的狀態相反  $Q_{n+1} = \overline{Q}_n$ 。因此，JK 正反器能當計數器，T 型正反器當然也行。以下將介紹以 T 型正反器完成計數器之方法。

表 4-2-4 T 型正反器真值表

T	CK	Q <sub>n+1</sub>
$\phi$	0	Q <sub>n</sub>
0	1	Q <sub>n</sub>
1	1	$\overline{Q}_n$

#### 1. MOD16 上數計數器

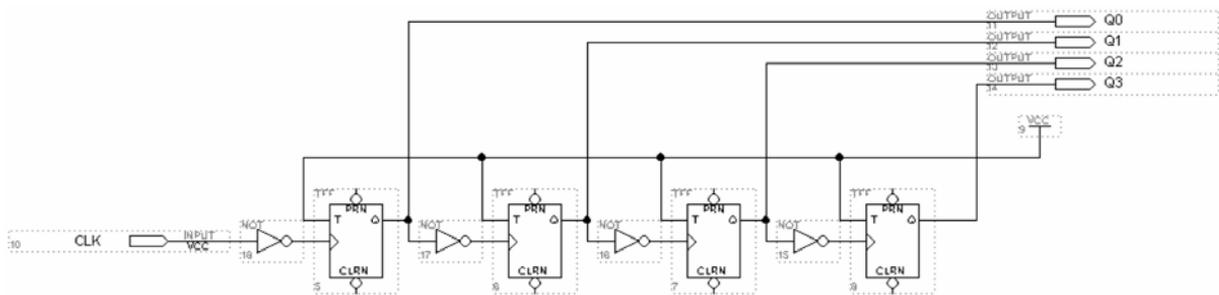


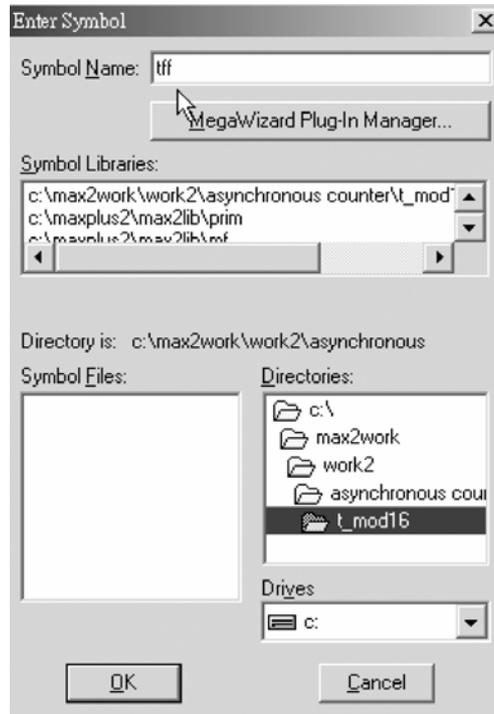
圖 4-2-25 MOD 16 上數計數器電路圖

如圖 4-2-25 所示經由步驟：畫電路圖→存檔→設為工作專案→指定 CPLD 晶片→編譯→軟體模擬→規劃腳位→編譯→下載燒錄(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

製作步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖，如圖 4-2-26 所示，利用 Symbol→Enter Symbol 叫出 Enter Symbol 視窗，輸入 tff 以叫出 T 型正反器(T flip-flop)。因

為此為正緣觸發，要做上數計數器需用負緣觸發，所以時脈輸入端前面需再接反閘(NOT)，依照前述方法畫出電路圖。



● 圖 4-2-26 元件取用視窗

2. 存檔，取檔名為 mod 16up.gdf。
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

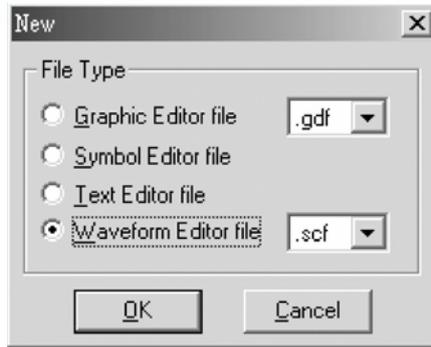
如圖 4-2-27 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



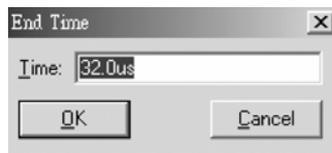
● 圖 4-2-27 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $32 \mu s$ ，設定格線間距(Options→Grid Size)設定  $1 \mu s$ ，顯示在視窗中適當大小格線(View→Fit in Window)。

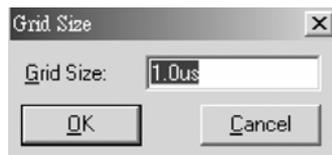
(因為在此計數器輸出有 4 個，可計數 0~15，總共有  $2^4=16$  種變化，當我們設 Grid= $1 \mu s$  時，一個時脈信號需要高態及低態各  $1 \mu s$ ，共需  $2 \mu s$ ，故計數 0~15 需 16 個時脈，總共需  $2 \mu s \times 16 = 32 \mu s$  才完成模擬。)



● 圖 4-2-28 開啟新檔視窗

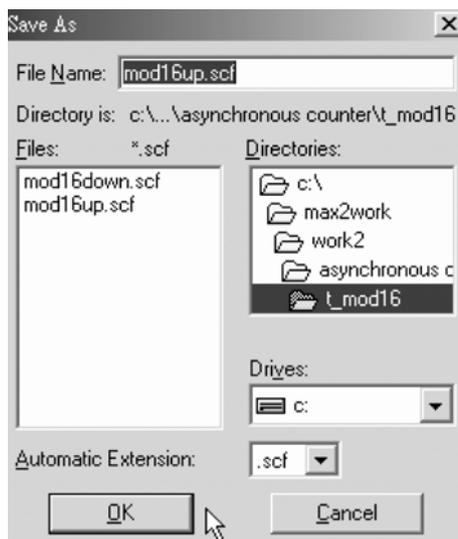


● 圖 4-2-29 模擬結束時間設定視窗

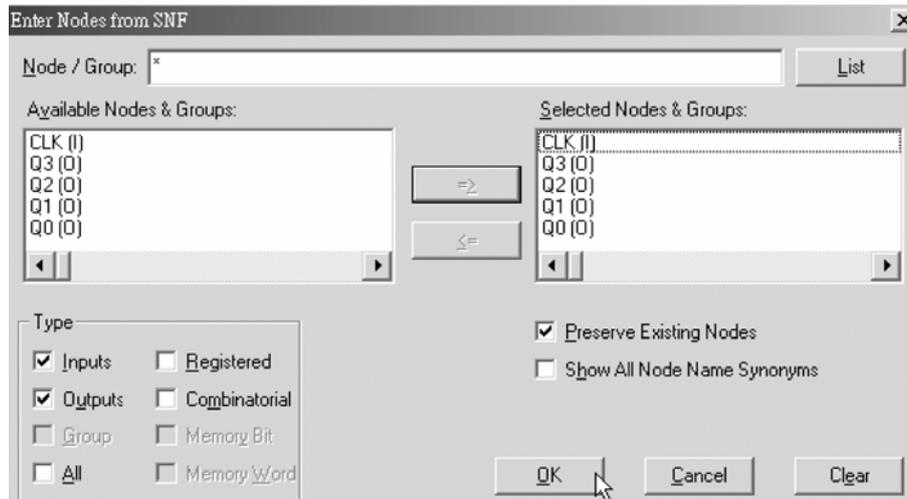


● 圖 4-2-30 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 mod 16up.scf，輸入節點(Node → Enter Nodes from SNF，按 List 及  $\Rightarrow$ ，OK)，編輯輸入信號，總共有三十二種組合，剛好 32  $\mu$ s 可模擬完成，可用計數時脈完成輸入信號編輯。

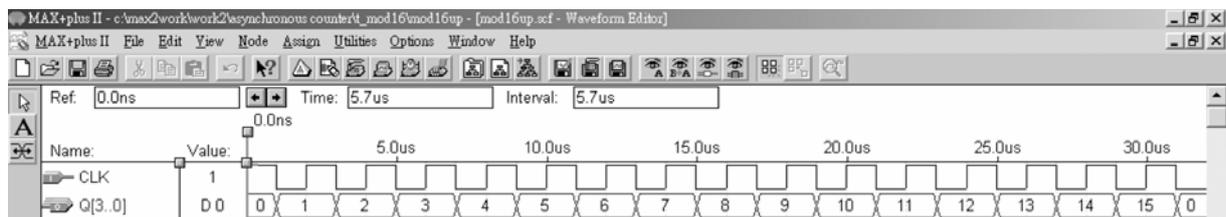


● 圖 4-2-31 儲存檔案視窗

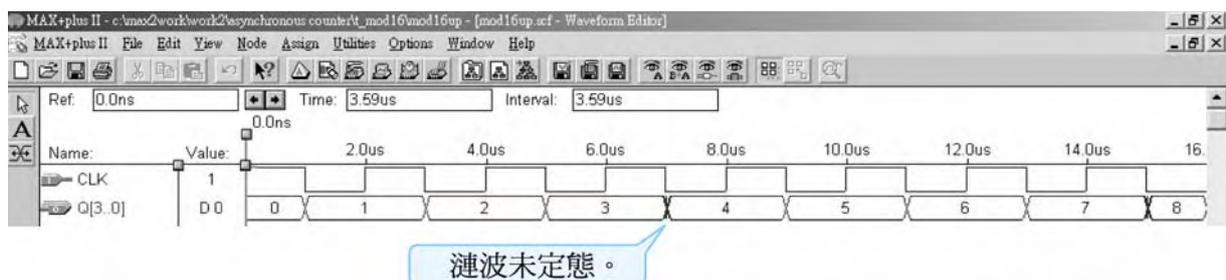


● 圖 4-2-32 輸出入節點選擇視窗

6. 執行模擬(MAX+PLUS II → Simulator, Start)，結果如圖 4-2-33 所示，可點選 Q[3..0]，按滑鼠右鍵選擇 Ungroup，可得各輸出端波形圖。模擬沒有錯誤及警告，所得波形模擬結果符合 MOD16 上數計數器，代表我們製作的電路是正確可用的。



● 圖 4-2-33 模擬結果視窗



● 圖 4-2-34 模擬結果視窗

另外補充一點：利用   調整大小，與上一單元的 JK 正反器之計數器相同情況，在有很多時間點有出現漣波未定態(如  $3\mu s$  時的上下振盪)，這是因為非同步計數器並非同時由同一個時脈輸入信號觸發，因此其會有不同正反器產生的延遲觸發間距的缺點，這種現象在同步計數器則不會出現。

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

表 4-2-5 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出			
名稱	CLK	Q0	Q1	Q2	Q3
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76
實驗器模組對應腳位	CLK <sub>1</sub>	DG <sub>0</sub>	DG <sub>1</sub>	DG <sub>2</sub>	DG <sub>3</sub>

實驗器可選擇使用 CLK<sub>1</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR<sub>1</sub> 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3 個綠色 LED 接腳當輸出端 Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>。

(2)MOD16 下數計數器

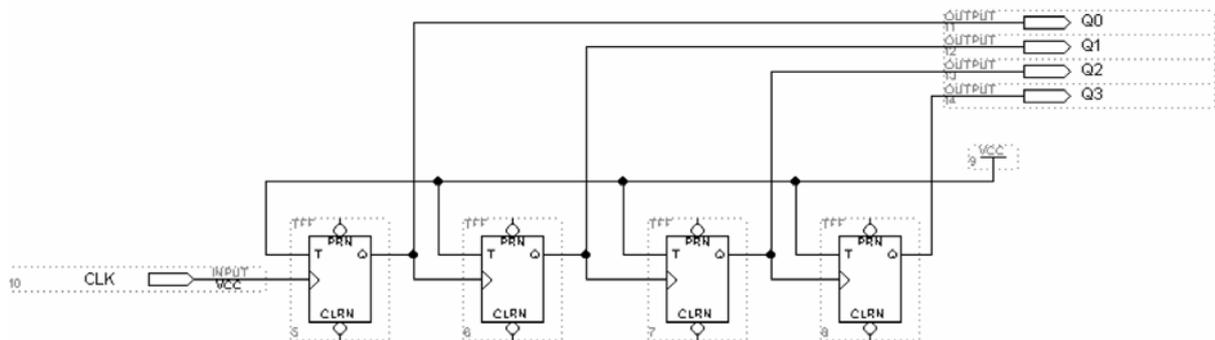
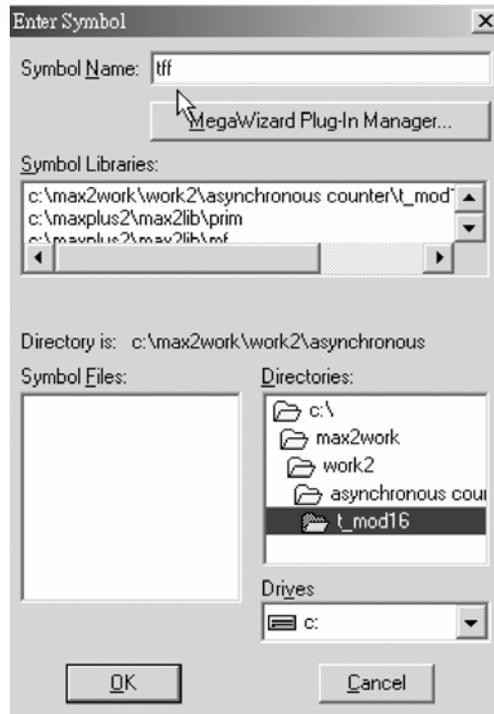


圖 4-2-35 MOD16 下數計數器電路圖

如圖 4-2-35 所示經由步驟：畫電路圖→存檔→設為工作專案→指定 CPLD 晶片→編譯→軟體模擬→規劃腳位→編譯→下載燒錄(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

製作步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖，如圖 4-2-36 所示，利用 Symbol→Enter Symbol 叫出 Enter Symbol 視窗，輸入 tff 以叫出 T 型正反器(T flip-flop)。因為此為正緣觸發，要做下數計數器需用正緣觸發，所以時脈輸入端前面不需如上數計數器接反閘(NOT)，依照前述方法畫出電路圖。



● 圖 4-2-36 元件取用視窗

2. 存檔，取檔名為 mod 16 down.gdf。
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

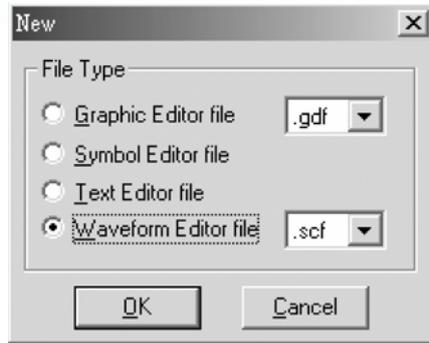
如圖 4-2-37 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



● 圖 4-2-37 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $32 \mu s$ ，設定格線間距(Options → Grid Size)設定  $1 \mu s$ ，顯示在視窗中適當大小格線(View→Fit in Window)。

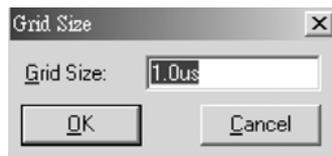
(因為在此計數器輸出有 4 個，可計數 0~15，總共有  $2^4=16$  種變化，當我們設 Grid= $1 \mu s$  時，一個時脈信號需要高態及低態各  $1 \mu s$ ，共需  $2 \mu s$ ，故計數 0~15 需 16 個時脈，總共需  $2 \mu s \times 16 = 32 \mu s$  才完成模擬。)



● 圖 4-2-38 開啟新檔視窗

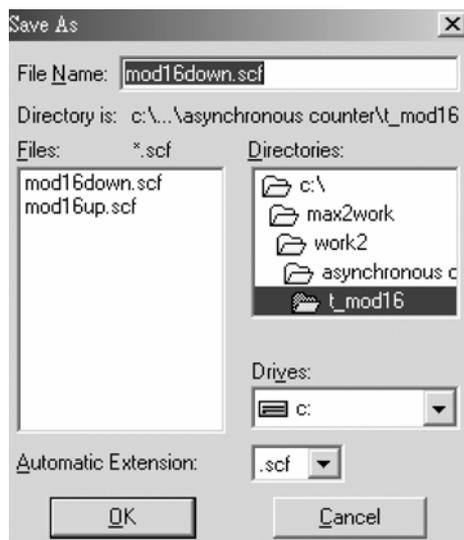


● 圖 4-2-39 模擬結束時間設定視窗

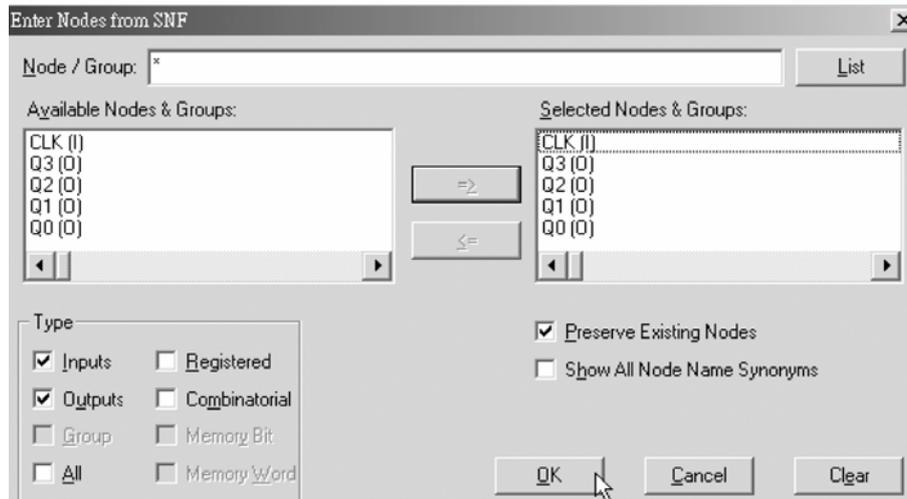


● 圖 4-2-40 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 mod 16up.scf，輸入節點(Node → Enter Nodes from SNF，按 List 及 ⇒，OK)，編輯輸入信號，總共有三十二種組合，剛好 32  $\mu$ s 可模擬完成，可用計數時脈完成輸入信號編輯。

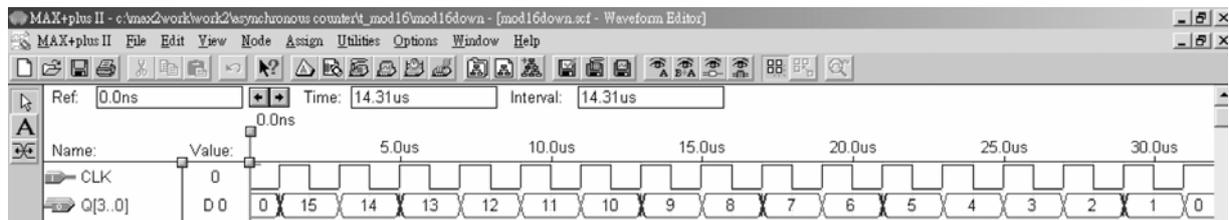


● 圖 4-2-41 儲存檔案視窗



● 圖 4-2-42 輸出入節點選擇視窗

6. 執行模擬(MAX+PLUS II → Simulator, Start)，結果如圖 4-2-43 所示，可點選 Q[3..0]，按滑鼠右鍵選擇 Ungroup，可得各輸出端波形圖。模擬沒有錯誤及警告，所得波形模擬結果符合 MOD16 下數計數器，代表我們製作的電路是正確可用的。



● 圖 4-2-43 模擬結果視窗

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

● 表 4-2-6 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出			
名稱	CLK	Q0	Q1	Q2	Q3
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76
實驗器模組對應腳位	CLK1	DG0	DG1	DG2	DG3

實驗器可選擇使用 CLK<sub>1</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR<sub>1</sub> 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3 個綠色 LED 接腳當輸出端 Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>。

## 4-3 同步計數器

正反器所做的計數器中，若是所有正反器的時脈輸入端是接在一起，由同一個時脈信號輸入各別的正反器中，則表示這些正反器是同時做輸出轉態動作，稱之為同步計數器(synchronous counter)，其總延遲時間為一個正反器的時間  $t_p$ ，其最大工作頻率  $f_{max} = 1/t_p$ 。同步計數器雖有速度較快之優點，其缺點則為硬體線路設計較為複雜。要讓正反器之輸出正常計數，則需對其輸入端設計適當輸入電路；在此可利用正反器之激勵表，在預知輸出的狀態下(即計數器的正常計數)，去推測該有怎樣的輸入，以下將分別以 JK 正反器及 D 型正反器為例介紹。

### 4-3-1 Jk 正反器 MOD16

要製作 MOD16 之同步計數器，則需有四個正反器  $2^4=16$ 。要完成 MOD16 上數計數器電路，可知其  $Q_3Q_2Q_1Q_0$  輸出依序為 0000,0001,0010,……1111(由 0~15 依序二進制計數)，配合 4-3-1 激勵表，可預知其輸入  $J_3K_3, J_2K_2, J_1K_1, J_0K_0$  如表 4-3-2 所示。

表 4-3-1 JK 正反器激勵表

$Q_n \rightarrow Q_{n+1}$	J	K
0→0	0	$\phi$
0→1	1	$\phi$
1→0	$\phi$	1
1→1	$\phi$	0

表 4-3-2 MOD 16 之同步計數器

計數	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>	J <sub>3</sub>	K <sub>3</sub>	J <sub>2</sub>	K <sub>2</sub>	J <sub>1</sub>	K <sub>1</sub>	J <sub>0</sub>	K <sub>0</sub>
0	0	0	0	0	0	φ	0	φ	0	φ	1	φ
1	0	0	0	1	0	φ	0	φ	1	φ	φ	1
2	0	0	1	0	0	φ	0	φ	φ	0	1	φ
3	0	0	1	1	0	φ	1	φ	φ	1	φ	1
4	0	1	0	0	0	φ	φ	0	0	φ	1	φ
5	0	1	0	1	0	φ	φ	0	1	φ	φ	1
6	0	1	1	0	0	φ	φ	0	φ	0	1	φ
7	0	1	1	1	1	φ	φ	1	φ	1	φ	1
8	1	0	0	0	φ	0	0	φ	0	φ	1	φ
9	1	0	0	1	φ	0	0	φ	1	φ	φ	1
10	1	0	1	0	φ	0	0	φ	φ	0	1	φ
11	1	0	1	1	φ	0	1	φ	φ	1	φ	1
12	1	1	0	0	φ	0	φ	0	0	φ	1	φ
13	1	1	0	1	φ	0	φ	0	1	φ	φ	1
14	1	1	1	0	φ	0	φ	0	φ	0	1	φ
15	1	1	1	1	φ	1	φ	1	φ	1	φ	1
0	0	0	0	0								

1. 由輸出是由 1→0，查激勵表以推測輸入 J<sub>0</sub>K<sub>0</sub> 應為 φ 1

2. 在 J<sub>0</sub>K<sub>0</sub> 填入 φ 1

四個 JK 正反器之輸入分別為 J<sub>3</sub>K<sub>3</sub>、J<sub>2</sub>K<sub>2</sub>、J<sub>1</sub>K<sub>1</sub>、J<sub>0</sub>K<sub>0</sub>。其輸出分別為 Q<sub>3</sub>Q<sub>2</sub>Q<sub>1</sub>Q<sub>0</sub>。其中 J<sub>3</sub>K<sub>3</sub> 的輸入影響 Q<sub>3</sub> 的輸出，J<sub>2</sub>K<sub>2</sub> 的輸入影響 Q<sub>2</sub> 的輸出，J<sub>1</sub>K<sub>1</sub> 的輸入影響 Q<sub>1</sub> 的輸出，J<sub>0</sub>K<sub>0</sub> 的輸入影響 Q<sub>0</sub> 的輸出。因此，可由激勵表來判斷 J<sub>3</sub>K<sub>3</sub>、J<sub>2</sub>K<sub>2</sub>、J<sub>1</sub>K<sub>1</sub>、J<sub>0</sub>K<sub>0</sub> 輸入應為何。由表可知，Q<sub>0</sub> 的輸出依序為 0→1，1→0，0→1，……，1→0。對照 JK 正反器之激勵表，可得 J<sub>0</sub>K<sub>0</sub> 之輸入依序為 1φ，φ1，1φ，……，φ1。Q<sub>1</sub> 的輸出依序為 0→0，0→1，1→1，1→0……，1→0。對照 JK 正反器之激勵表，可得 J<sub>1</sub>K<sub>1</sub> 之輸入依序為 0φ，1φ，φ0，φ1，……，φ1。Q<sub>2</sub> 的輸出依序為 0→0，0→0，0→0，0→1……，1→0。對照 JK 正反器之激勵表，可得 J<sub>2</sub>K<sub>2</sub> 之輸入依序為 0φ，0φ，0φ，1φ，……，φ1。Q<sub>3</sub> 的輸出依序為 0→0，0→0，0→0……，1→0。對照 JK 正反器之激勵表，J<sub>3</sub>K<sub>3</sub> 可得之輸入依序為 0φ，0φ，φ0，0φ，……，φ1。應用此方式，依序填完 0~15 之計數各狀態。

表 4-3-3 MOD16 之同步計數器

計數	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>	J <sub>3</sub>	K <sub>3</sub>	J <sub>2</sub>	K <sub>2</sub>	J <sub>1</sub>	K <sub>1</sub>	J <sub>0</sub>	K <sub>0</sub>
0	0	0	0	0	0	φ	0	φ	0	φ	1	φ
1	0	0	0	1	0	φ	0	φ	1	φ	φ	1
2	0	0	1	0	0	φ	0	φ	φ	0	1	φ
3	0	0	1	1	0	φ	1	φ	φ	1	φ	1
4	0	1	0	0	0	φ	φ	0	0	φ	1	φ
5	0	1	0	1	0	φ	φ	0	1	φ	φ	1
6	0	1	1	0	0	φ	φ	0	φ	0	1	φ
7	0	1	1	1	1	φ	φ	1	φ	1	φ	1
8	1	0	0	0	φ	0	0	φ	0	φ	1	φ
9	1	0	0	1	φ	0	0	φ	1	φ	φ	1
10	1	0	1	0	φ	0	0	φ	φ	0	1	φ
11	1	0	1	1	φ	0	1	φ	φ	1	φ	1
12	1	1	0	0	φ	0	φ	0	0	φ	1	φ
13	1	1	0	1	φ	0	φ	0	1	φ	φ	1
14	1	1	1	0	φ	0	φ	0	φ	0	1	φ
15	1	1	1	1	φ	1	φ	1	φ	1	φ	1
0	0	0	0	0								

由表 4-3-3 經由卡諾圖分別化簡 J<sub>3</sub>、K<sub>3</sub>；J<sub>2</sub>、K<sub>2</sub>；J<sub>1</sub>、K<sub>1</sub>；J<sub>0</sub>、K<sub>0</sub> 所得之布林代數如下所示：

$$J_3 = Q_2 Q_1 Q_0$$

$$K_3 = Q_2 Q_1 Q_0$$

$$J_2 = Q_1 Q_0$$

$$K_2 = Q_1 Q_0$$

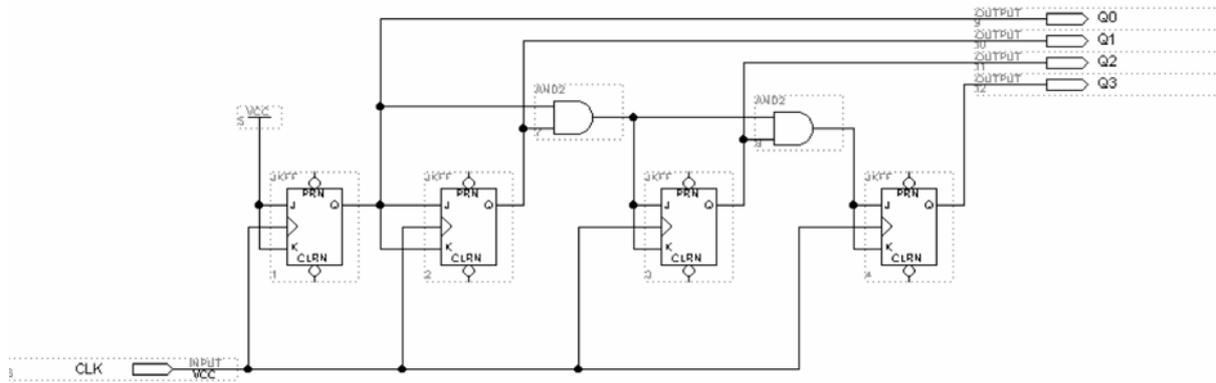
$$J_1 = Q_0$$

$$K_1 = Q_0$$

$$J_0 = 1$$

$$K_0 = 1$$

有了四個正反器的各輸入端之布林函數，便可依此繪製電路圖。



● 圖 4-3-1 MOD 16 同步上數計數器電路圖

如圖 4-3-1 所示經由步驟：**畫電路圖**→**存檔**→**設為工作專案**→**指定 CPLD 晶片**→**編譯**→**軟體模擬**→**規劃腳位**→**編譯**→**下載燒錄**(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

**製作 MOD16 同步上數計數器步驟如下：**

1. 開啟新圖形編輯檔，繪製邏輯電路圖，如圖 4-3-1 所示，將所有正反器的輸入時脈接在一起，再接到一個時脈輸入端，因為  $J_2 = Q_1Q_0 = K_2$ ，所以將  $Q_1Q_0$  接到一個 AND 閘再接到  $J_2$  及  $K_2$ ，而  $J_3 = Q_2Q_1Q_0 = K_3$ ，因此可將  $J_2 (= Q_1Q_0)$  及  $Q_2$  接到一個 AND 閘再接到  $J_3$  及  $K_3$ ，其它依述前面介紹方法畫出電路圖。
2. 存檔，取檔名為 mod 16 up.gdf
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

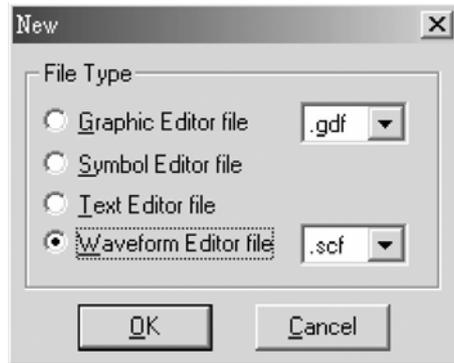
如圖 4-3-2 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



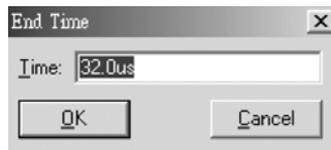
● 圖 4-3-2 編譯訊息視窗

- 4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定 32  $\mu$ s，設定格線間距(Options→Grid Size)設定 1  $\mu$ s，顯示在視窗中適當大小格線(View→Fit in Window)。

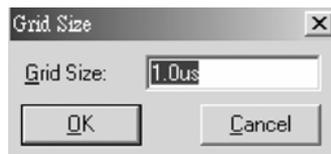
(因為在此計數器輸出有 4 個，可計數 0~15，總共有  $2^4=16$  種變化，當我們設 Grid=1  $\mu$ s 時，一個時脈信號需要高態及低態各 1  $\mu$ s，共需 2  $\mu$ s，故計數 0~15 需 16 個時脈，總共需 2  $\mu$ s  $\times$  16= 32  $\mu$ s 才完成模擬。)



● 圖 4-3-3 開啟新檔視窗

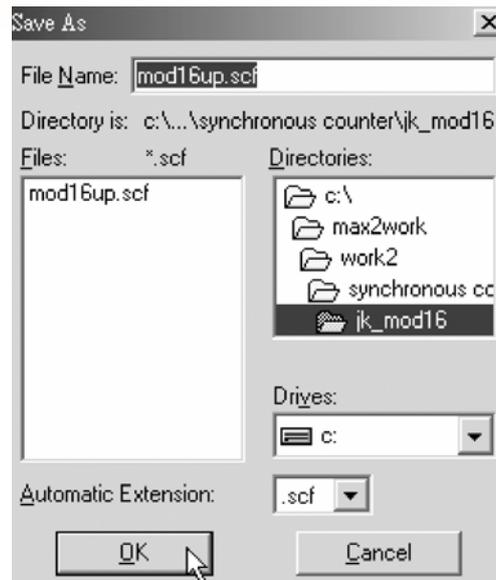


● 圖 4-3-4 模擬結束時間設定視窗

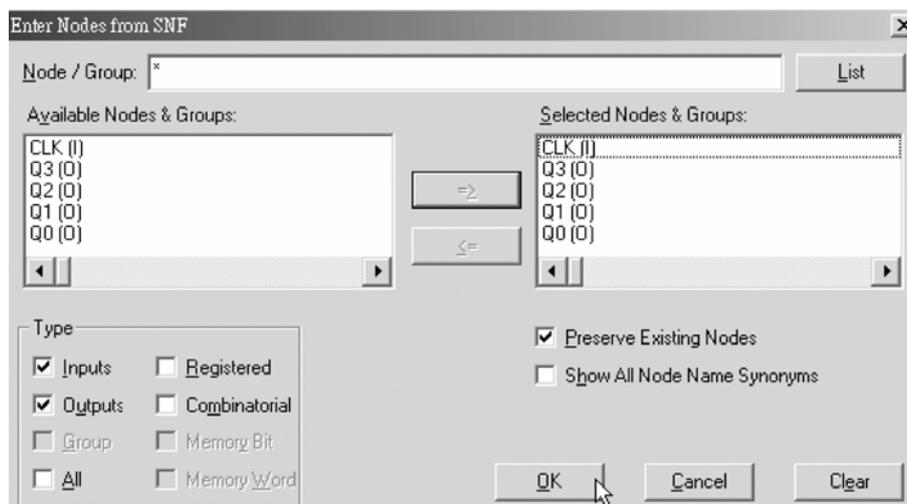


● 圖 4-3-5 模擬單位時間設定視窗

- 5. 儲存檔案(Save As)，檔名 mod 16up.scf，輸入節點(Node → Enter Nodes from SNF，按 List 及 =>，OK)，編輯輸入信號，總共有三十二種組合，剛好 32  $\mu$ s 可模擬完成，可用計數時脈完成輸入信號編輯。

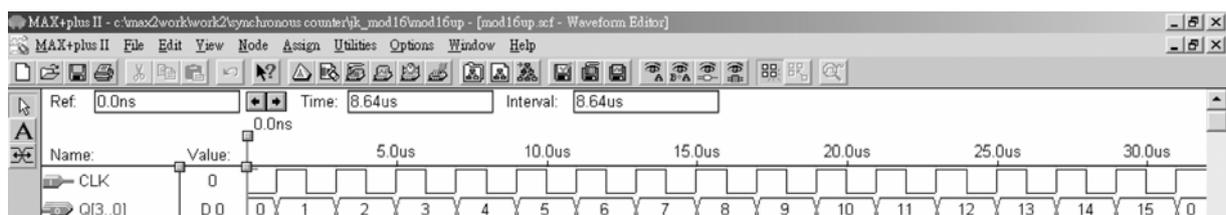


● 圖 4-3-6 儲存檔案視窗



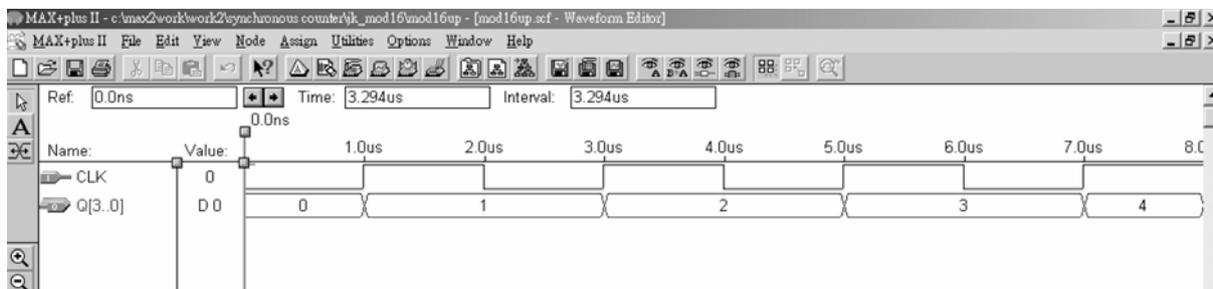
● 圖 4-3-7 輸出入節點選擇視窗

6. 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-3-8 所示，可得各輸出端波形圖。模擬沒有錯誤及警告，所得波形模擬結果符合 MOD16 上數計數器，代表我們製作的電路是正確可用的。



● 圖 4-3-8 模擬結果視窗

另外補充一點：利用   調整大小，與上一單元之非同步計數器比較下，我們可知在同步計數器的輸出中，並未發現在非同步計數器出現的漣波未定態，這是因為同步計數器是同時由同一個時脈輸入信號觸發，因此不會有不同正反器產生的延遲觸發間距的缺點，這也是同步計數器的優點之一。



● 圖 4-3-9 模擬結果視窗

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

● 表 4-3-4 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出			
名稱	CLK	Q0	Q1	Q2	Q3
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76
實驗器模組對應腳位	CLK1	DG0	DG1	DG2	DG3

實驗器可選擇使用 CLK<sub>1</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR1 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3 個綠色 LED 接腳當輸出端 Q<sub>0</sub>，Q<sub>1</sub>，Q<sub>2</sub>，Q<sub>3</sub>。

### 4-3-2 JK 正反器 MOD10

要製作 MOD10 之同步計數器，需有四個正反器，因為  $2^3=8$ ，使用三個正反器最多只能有八種狀態，無法表示 0~9 的十種狀態(MOD10)，而  $2^4=16$ ，可以計數 0~15，可以選擇 10~15 的六種狀態不使用，因此可用來完成計數 0~9 的 MOD10 計數器。要完成 MOD10 上數計數器電路，可知其 Q<sub>3</sub>Q<sub>2</sub>Q<sub>1</sub>Q<sub>0</sub> 輸出依序為 0000,0001,0010,...1001(由 0~9 依序二進制計數)，配合表 4-3-5 激勵表，可知其輸入 J<sub>3</sub>K<sub>3</sub>，J<sub>2</sub>K<sub>2</sub>，J<sub>1</sub>K<sub>1</sub>，J<sub>0</sub>K<sub>0</sub> 應如表 4-3-6 所示。

● 表 4-3-5 JK 正反器激勵表

$Q_n \rightarrow Q_{n+1}$	J	K
0→0	0	$\phi$
0→1	1	$\phi$
1→0	$\phi$	1
1→1	$\phi$	0

● 表 4-3-6 MOD10 之同步計數器

計數	$Q_3$	$Q_2$	$Q_1$	$Q_0$	$J_3$	$K_3$	$J_2$	$K_2$	$J_1$	$K_1$	$J_0$	$K_0$
0	0	0	0	0	0	$\phi$	0	$\phi$	0	$\phi$	1	$\phi$
1	0	0	0	1	0	$\phi$	0	$\phi$	1	$\phi$	$\phi$	1
2	0	0	1	0	0	$\phi$	0	$\phi$	$\phi$	0	1	$\phi$
3	0	0	1	1	0	$\phi$	1	$\phi$	$\phi$	1	$\phi$	1
4	0	1	0	0	0	$\phi$	$\phi$	0	0	$\phi$	1	$\phi$
5	0	1	0	1	0	$\phi$	$\phi$	0	1	$\phi$	$\phi$	1
6	0	1	1	0	0	$\phi$	$\phi$	0	$\phi$	0	1	$\phi$
7	0	1	1	1	1	$\phi$	$\phi$	1	$\phi$	1	$\phi$	1
8	1	0	0	0	$\phi$	0	0	$\phi$	0	$\phi$	1	$\phi$
9	1	0	0	1	$\phi$	1	0	$\phi$	0	$\phi$	$\phi$	1
0	0	0	0	0								

四個 JK 正反器之輸入分別為  $J_3K_3$ 、 $J_2K_2$ 、 $J_1K_1$ 、 $J_0K_0$ 。其輸出分別為  $Q_3Q_2Q_1Q_0$ 。其中  $J_3K_3$  的輸入影響  $Q_3$  的輸出， $J_2K_2$  的輸入影響  $Q_2$  的輸出， $J_1K_1$  的輸入影響  $Q_1$  的輸出， $J_0K_0$  的輸入影響  $Q_0$  的輸出。因此，可由激勵表來判斷  $J_3K_3$ 、 $J_2K_2$ 、 $J_1K_1$ 、 $J_0K_0$  輸入應為何。由表可知， $Q_0$  的輸出依序為  $0 \rightarrow 1$ ， $1 \rightarrow 0$ ， $0 \rightarrow 1$ ，...， $1 \rightarrow 0$ 。對照 JK 正反器之激勵表，可得  $J_0K_0$  之輸入依序為  $1\phi$ ， $\phi 1$ ， $1\phi$ ，...， $\phi 1$ 。 $Q_1$  的輸出依序為  $0 \rightarrow 0$ ， $0 \rightarrow 1$ ， $1 \rightarrow 1$ ， $1 \rightarrow 0$ ...， $0 \rightarrow 0$ 。對照 JK 正反器之激勵表，可得  $J_1K_1$  之輸入依序為  $0\phi$ ， $1\phi$ ， $\phi 0$ ， $\phi 1$ ，...， $0\phi$ 。 $Q_2$  的輸出依序為  $0 \rightarrow 0$ ， $0 \rightarrow 0$ ， $0 \rightarrow 0$ ， $0 \rightarrow 1$ ...， $0 \rightarrow 0$ 。對照 JK 正反器之激勵表，可得  $J_2K_2$  之輸入依序為  $0\phi$ ， $0\phi$ ， $0\phi$ ， $1\phi$ ，...， $0\phi$ 。 $Q_3$  的輸出依序為  $0 \rightarrow 0$ ， $0 \rightarrow 0$ ， $0 \rightarrow 0$ ， $0 \rightarrow 0$ ...， $1 \rightarrow 0$ 。對照 JK 正反器之激勵表， $J_3K_3$  可得之輸入依序為  $0\phi$ ， $0\phi$ ， $\phi 0$ ， $0\phi$ ，...， $\phi 1$ 。應用此方式，依序填完 0~9 之計數各狀態，而不會出現的(計數 10~15)狀態，在化簡過程中視為  $\phi$  (don't care)。

由表 4-3-6 經由卡諾圖分別化簡  $J_3$ 、 $K_3$ ； $J_2$ 、 $K_2$ ； $J_1$ 、 $K_1$ ； $J_0$ 、 $K_0$  後可得布林代數如下所示：

$$\begin{aligned}
 J_3 &= Q_2 Q_1 Q_0 \\
 K_3 &= Q_0 \\
 J_2 &= Q_1 Q_0 \\
 K_2 &= Q_1 Q_0 \\
 J_1 &= \overline{Q}_3 Q_0 \\
 K_1 &= \overline{Q}_3 Q_0 \\
 J_0 &= 1 \\
 K_0 &= 1
 \end{aligned}$$

有了四個正反器的各輸入端之布林函數，便可依此繪製電路圖。

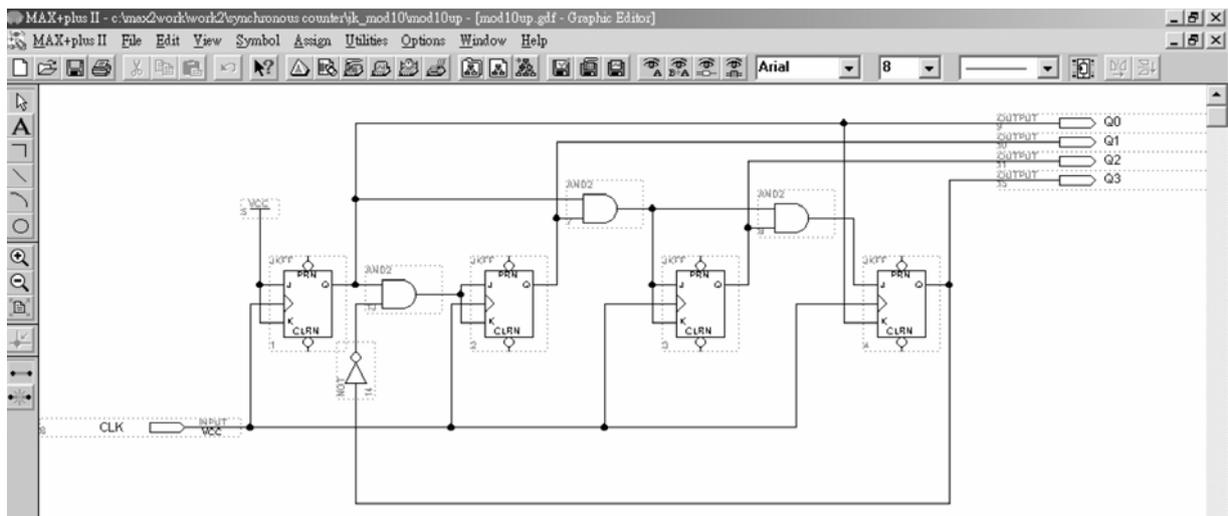


圖 4-3-10 MOD10 同步上數計數器電路圖

如圖 4-3-10 所示經由步驟：畫電路圖→存檔→設為工作專案→指定 CPLD 晶片→編譯→軟體模擬→規劃腳位→編譯→下載燒錄(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

製作 MOD10 同步上數計數器步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖。
2. 存檔，取檔名為 mod 10up.gdf
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

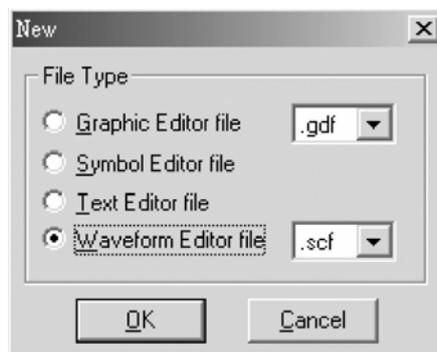
如圖 4-3-11 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



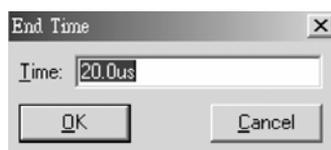
● 圖 4-3-11 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $20\ \mu\text{s}$ ，設定格線間距(Options → Grid Size)設定  $1\ \mu\text{s}$ ，顯示在視窗中適當大小格線(View→Fit in Window)。

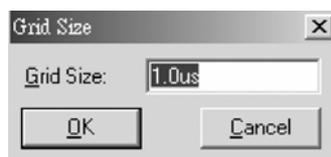
(因為在此每一個時脈計數一次，一個時脈需有  $2\ \mu\text{s}$ (高態及低態各  $1\ \mu\text{s}$ )，計數 0~9，總共有 20 種變化，所以需  $20\ \mu\text{s}$ 。)



● 圖 4-3-12 開啟新檔視窗

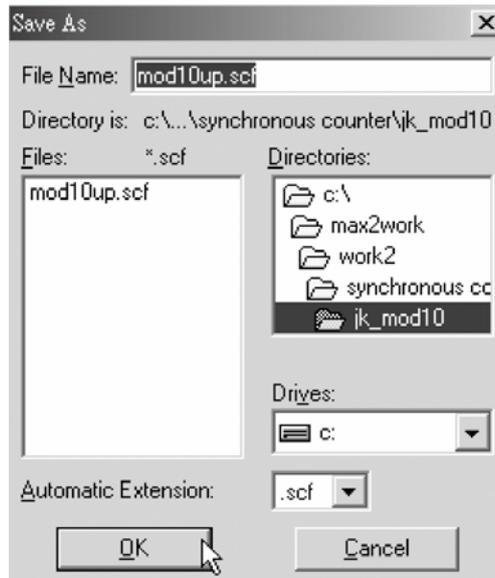


● 圖 4-3-13 模擬結束時間設定視窗

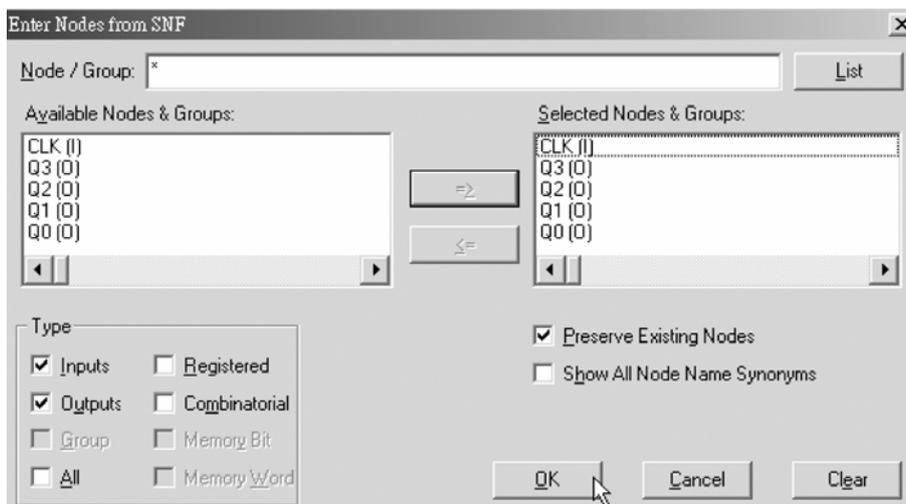


● 圖 4-3-14 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 mod 10up.scf，輸入節點(Node→Enter Nodes from SNF，按 List 及  $\Rightarrow$ ，OK)，編輯輸入信號，可用計數時脈完成輸入信號編輯。

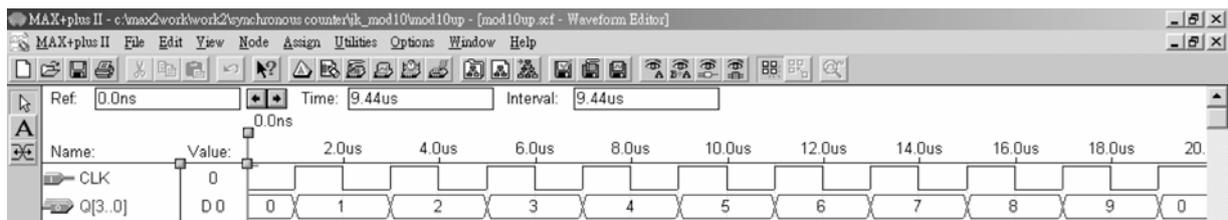


● 圖 4-3-15 儲存檔案視窗



● 圖 4-3-16 輸出入節點選擇視窗

- 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-3-17 所示，可得各輸出端波形圖。模擬沒有錯誤及警告，所得波形模擬結果符合 MOD10 上數計數器，代表我們製作的電路是正確可用的。



● 圖 4-3-17 模擬結果視窗

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

● 表 4-3-7 電路圖輸出腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出			
名稱	CLK	Q0	Q1	Q2	Q3
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76
實驗器模組對應腳位	CLK1	DG0	DG1	DG2	DG3

實驗器可選擇使用 CLK<sub>1</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR1 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3 個綠色 LED 接腳當輸出端 Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>。

#### 4-3-3 D 型正反器 MOD16

除了 JK 正反器可製作同步計數器，D 型正反器亦可。以 D 型正反器製作 MOD16 之同步計數器，同樣需有四個正反器  $2^4=16$ 。要完成 MOD16 上數計數器電路，可知其 Q<sub>3</sub>Q<sub>2</sub>Q<sub>1</sub>Q<sub>0</sub> 輸出依序為 0000,0001,0010,...,1111(由 0~15 依序二進制計數)，配合激勵表，可預知輸入 D<sub>3</sub>D<sub>2</sub>D<sub>1</sub>D<sub>0</sub> 如表所示。

● 表 4-3-8 D 型正反器激勵表

Q <sub>n</sub> → Q <sub>n+1</sub>	D
0 → 0	0
0 → 1	1
1 → 0	0
1 → 1	1

四個 D 型正反器之輸入分別為 D<sub>3</sub>D<sub>2</sub>D<sub>1</sub>D<sub>0</sub>。其輸出分別為 Q<sub>3</sub>Q<sub>2</sub>Q<sub>1</sub>Q<sub>0</sub>。其中 D<sub>3</sub> 的輸入影響 Q<sub>3</sub> 的輸出，D<sub>2</sub> 的輸入影響 Q<sub>2</sub> 的輸出，D<sub>1</sub> 的輸入影響 Q<sub>1</sub> 的輸出，D<sub>0</sub> 的輸入影響 Q<sub>0</sub> 的輸出。因此，可由激勵表來判斷 D<sub>3</sub>D<sub>2</sub>D<sub>1</sub>D<sub>0</sub> 輸入應為何。由表可知，Q<sub>0</sub> 的輸出依序為 0 → 1, 1 → 0, 0 → 1, ..., 1 → 0。對照 D 型正反器之激勵表，可得 D<sub>0</sub> 之輸入依序為 1, 0, 1, ..., 0。Q<sub>1</sub> 的輸出依序為 0 → 0, 0 → 1, 1 → 1, 1 → 0, ..., 1 → 0。對照 D 型正反器之激勵表，可得 D<sub>1</sub> 之輸入依序為 0, 1, 1, 0, ..., 0。Q<sub>2</sub> 的輸出依序為 0 → 0, 0 → 0, 0 → 0, 0 → 1, ..., 1 → 0。對照 D 型正反器之激勵表，可得 D<sub>2</sub> 之輸入依序為 0, 0, 0, 1, ..., 0。Q<sub>3</sub> 的輸出依序為

0→0, 0→0, 0→0, 0→0..., 1→0。對照 D 型正反器之激勵表，D<sub>3</sub> 可得之輸入依序為 0, 0, 0, 0, ..., 0。應用此方式，依序填完 0~15 之計數各狀態。

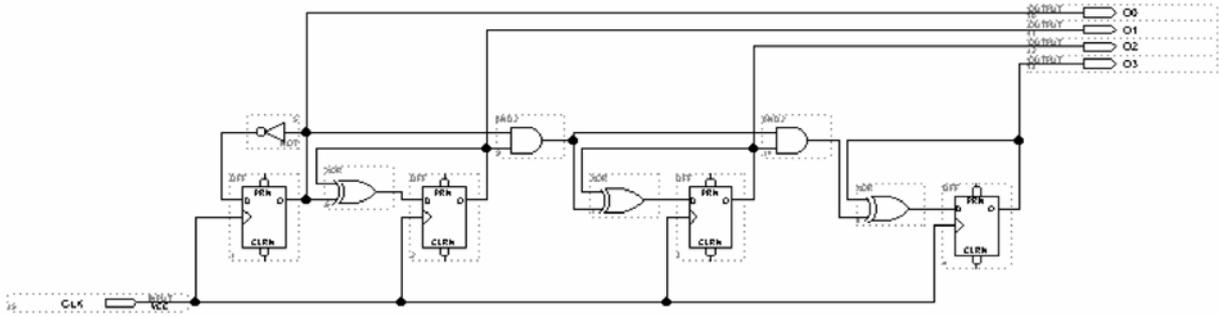
表 4-3-9 D 型正反器之 MOD 16 同步計數器

計數	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
0	0	0	0	0	0	0	0	1
1	0	0	0	1	0	0	1	0
2	0	0	1	0	0	0	1	1
3	0	0	1	1	0	1	0	0
4	0	1	0	0	0	1	0	1
5	0	1	0	1	0	1	1	0
6	0	1	1	0	0	1	1	1
7	0	1	1	1	1	0	0	0
8	1	0	0	0	1	0	0	1
9	1	0	0	1	1	0	1	0
10	1	0	1	0	1	0	1	1
11	1	0	1	1	1	1	0	0
12	1	1	0	0	1	1	0	1
13	1	1	0	1	1	1	1	0
14	1	1	1	0	1	1	1	1
15	1	1	1	1	0	0	0	0
0	0	0	0	0				

由表 4-3-9 經由卡諾圖化簡後可得 D<sub>3</sub>、D<sub>2</sub>、D<sub>1</sub>、D<sub>0</sub> 之布林代數如下所示：

$$\begin{aligned}
 D_3 &= (Q_2Q_1Q_0) \oplus Q_3 \\
 D_2 &= Q_1Q_0 \oplus Q_2 \\
 D_1 &= Q_0 \oplus Q_1 \\
 D_0 &= \overline{Q_0}
 \end{aligned}$$

有了四個正反器的各輸入端之布林函數，便可依此繪製電路圖。



● 圖 4-3-18 MOD 16 同步上數計數器電路圖

如圖 4-3-18 所示經由步驟：畫電路圖→存檔→設為工作專案→指定 CPLD 晶片→編譯→軟體模擬→規劃腳位→編譯→下載燒錄(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

製作 MOD10 同步上數計數器步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖，如圖 4-3-18 所示，將所有正反器的輸入時脈接在一起，再接到一個時脈輸入端，因為  $J_2 = Q_1Q_0 = K_2$ ，所以將  $Q_1Q_0$  接到一個 AND 閘再接到  $J_2$  及  $K_2$ ，而  $J_3 = Q_2Q_1Q_0 = K_3$ ，因此可將  $J_2 (= Q_1Q_0)$  及  $Q_2$  接到一個 AND 閘再接到  $J_3$  及  $K_3$ ，其它依述前面介紹方法畫出電路圖。
2. 存檔，取檔名為 mod 16up.gdf
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

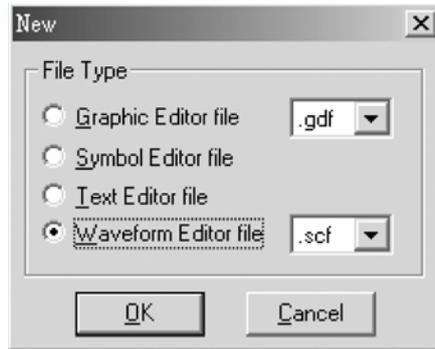
如圖 4-3-19 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



● 圖 4-3-19 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $32 \mu s$ ，設定格線間距(Options→Grid Size)設定  $1 \mu s$ ，顯示在視窗中適當大小格線(View→Fit in Window)。

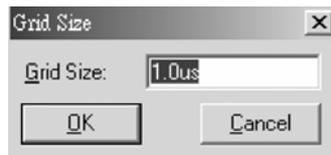
(因為在此計數器輸出有 4 個，可計數 0~15，總共有  $2^4 = 16$  種變化，當我們設 Grid= $1 \mu s$  時，一個時脈信號需要高態及低態各  $1 \mu s$ ，共需  $2 \mu s$ ，故計數 0~15 需 16 個時脈，總共需  $2 \mu s \times 16 = 32 \mu s$  才完成模擬。)



● 圖 4-3-20 開啟新檔視窗

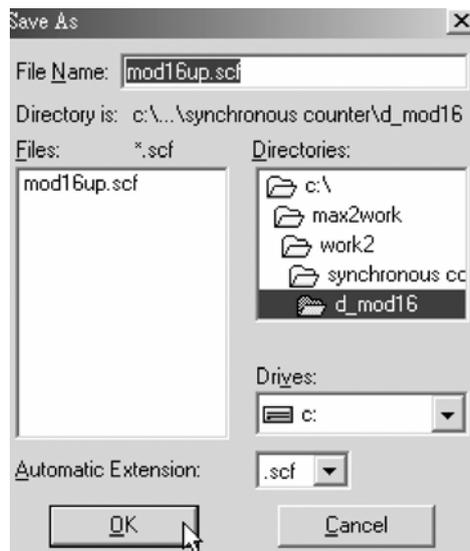


● 圖 4-3-21 模擬結束時間設定視窗

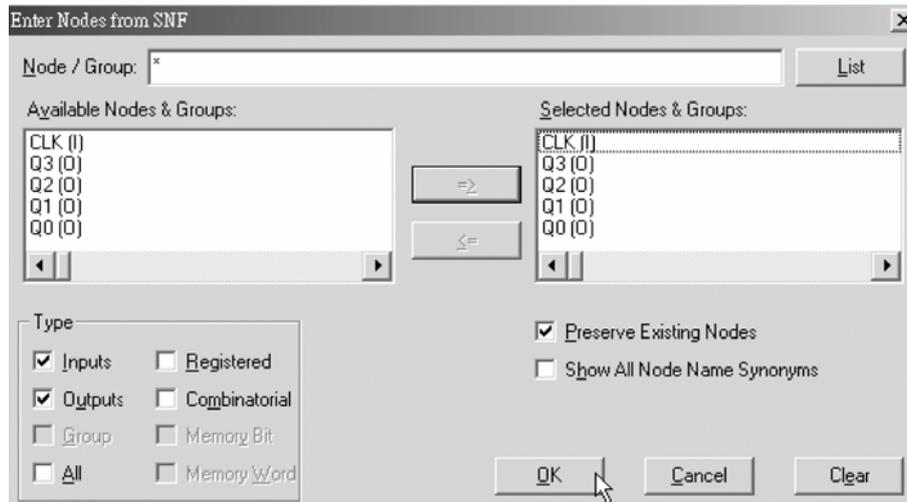


● 圖 4-3-22 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 mod 16up.scf，輸入節點(Node→Enter Nodes from SNF，按 List 及 ⇒，OK)，編輯輸入信號，總共有三十二種組合，剛好 32  $\mu$ s 可模擬完成，可用計數時脈完成輸入信號編輯。

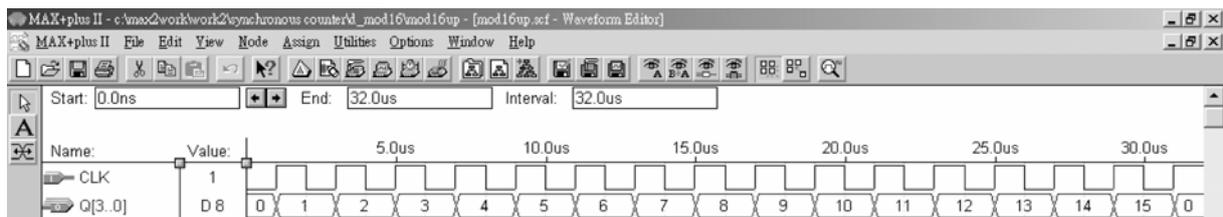


● 圖 4-3-23 儲存檔案視窗



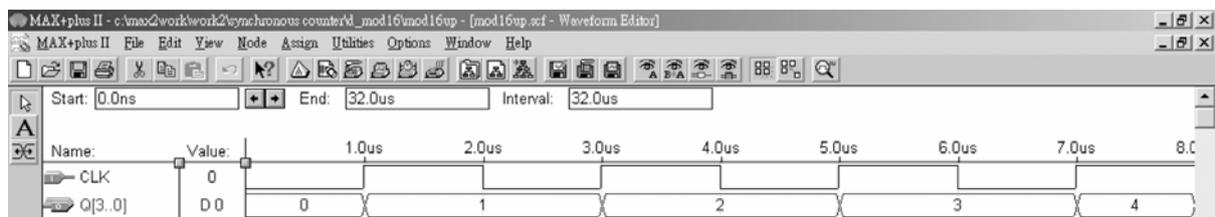
● 圖 4-3-24 輸出入節點選擇視窗

6. 執行模擬(MAX+PLUS II → Simulator, Start)，結果如圖 4-3-25 所示，可得各輸出端波形圖。模擬沒有錯誤及警告，所得波形模擬結果符合 MOD16 上數計數器，代表我們製作的電路是正確可用的。



● 圖 4-3-25 模擬結果視窗

另外補充一點：利用 調整大小，與上一單元之 JK 正反器做的同步計數器比較下，我們可知一樣是同步計數器，不管是 D 型或 JK 正反器所完成，皆不會如非同步計數器出現漣波未定態的現象。因為只要是同步計數器，皆是同時由同一個時脈輸入信號觸發，因此不會有不同正反器產生的延遲觸發間距的缺點，這與用什麼正反器所製作的計數器無關。



● 圖 4-3-26 模擬結果視窗

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

表 4-3-10 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出			
名稱	CLK	Q0	Q1	Q2	Q3
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76
實驗器模組對應腳位	CLK1	DG0	DG1	DG2	DG3

實驗器可選擇使用 CLK<sub>1</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR<sub>1</sub> 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3 個綠色 LED 接腳當輸出端 Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>。

#### 4-3-4 D 型正反器 MOD 32768

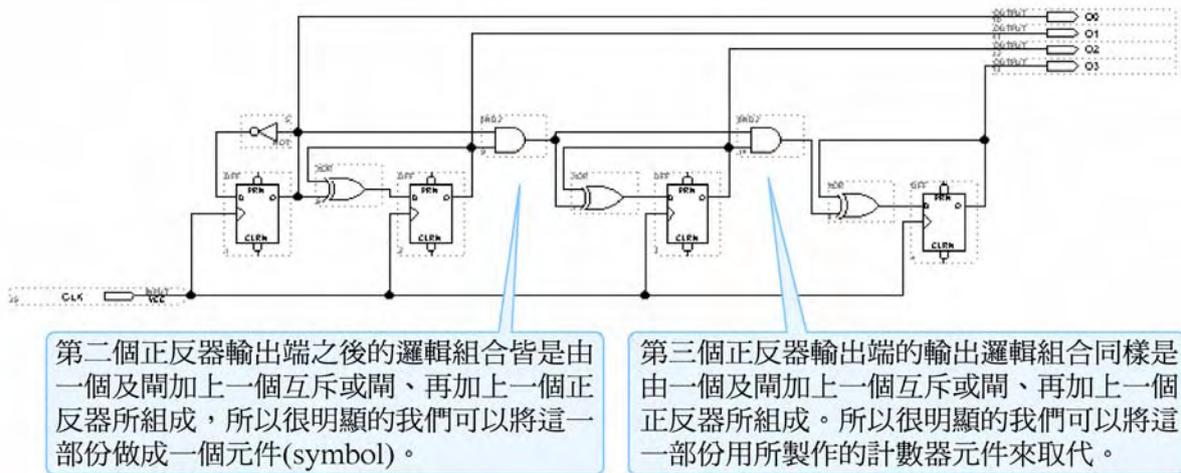
使用正反器可以製作計數器，只要正反器夠多，要計數到多大應該都沒有問題。理論上雖是如此，但若要計數到很大的數字，使用的正反器數目一多，要設計還是不容易。例如要設計 MOD32768 的計數器，光是真值表要寫 32768 種狀態，畫起來的圖表是相當驚人，所以若是按照上述方法去做是很差的做法。因此，應該觀察計數狀態少的計數器以找出規則，再照規則去做，則可得較為簡易之方法。如下所示為使用 D 型正反器製作之 MOD16 電路之各輸入端之布林函數。

$$\begin{aligned} D_3 &= (Q_2 Q_1 Q_0) \oplus Q_3 \\ D_2 &= Q_1 Q_0 \oplus Q_2 \\ D_1 &= Q_0 \oplus Q_1 \\ D_0 &= \overline{Q_0} \end{aligned}$$

由上可知若要設計 MOD 2 同步計數器可使用一個 D 型正反器，且輸入  $D_0 = \overline{Q_0}$ ；若要設計 MOD 4 同步計數器可使用二個 D 型正反器，且輸入  $D_0 = \overline{Q_0}$ ， $D_1 = Q_0 \oplus Q_1$ ；若要設計 MOD 8 同步計數器可使用三個 D 型正反器，且輸入  $D_0 = \overline{Q_0}$ ， $D_1 = Q_0 \oplus Q_1$ ， $D_2 = Q_1 Q_0 \oplus Q_2$ ；若要設計 MOD16 同步計數器可使用四個 D 型正反器，且輸入  $D_0 = \overline{Q_0}$ ； $D_1 = Q_0 \oplus Q_1$ ， $D_2 = Q_1 Q_0 \oplus Q_2$ ， $D_3 = (Q_2 Q_1 Q_0) \oplus Q_3$ 。依此類推，可知若要設計 MOD 32 同步計數器可使用五個 D 型正反器，且輸入  $D_0 = \overline{Q_0}$ ； $D_1 = Q_0 \oplus Q_1$ ， $D_2 = Q_1 Q_0 \oplus Q_2$ ， $D_3 = (Q_2 Q_1 Q_0) \oplus Q_3$ ， $D_4 = (Q_3 Q_2 Q_1 Q_0) \oplus Q_3$ 。以此方法類推，可得使用 n 個 D 型正反器製作 MOD 2<sup>n</sup>

同步計數器之各輸入端布林函數。

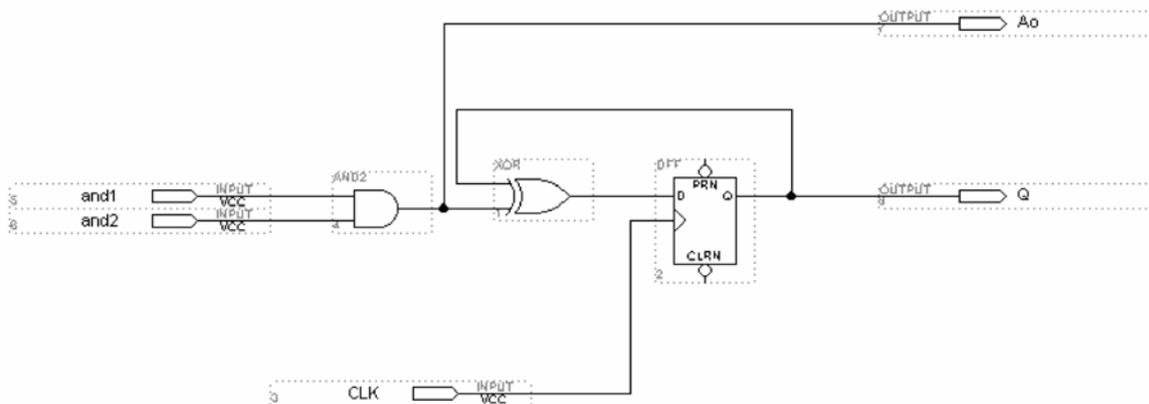
在電路方面，第一個 D 型正反器可直接由輸出端接一反閘拉回輸入端以及經由互斥或閘輸入到第二個正反器；第二個正反器輸出與第一個正反器輸出經由及閘送入第三個正反器輸入端前的互斥或閘。由圖 4-3-27 可觀察得知，第二個正反器輸出端之後的邏輯組合皆是由一個及閘加上一個互斥或閘、再加上一個正反器所組成，所以很明顯的我們可以將這一部份做成一個元件(symbol)，製作正反器之計數器時，重覆使用該元件即可得到高計數值之計數器，以下將以 MOD<sub>32768</sub> 為例。



● 圖 4-3-27 MOD16 同步上數計數器電路圖

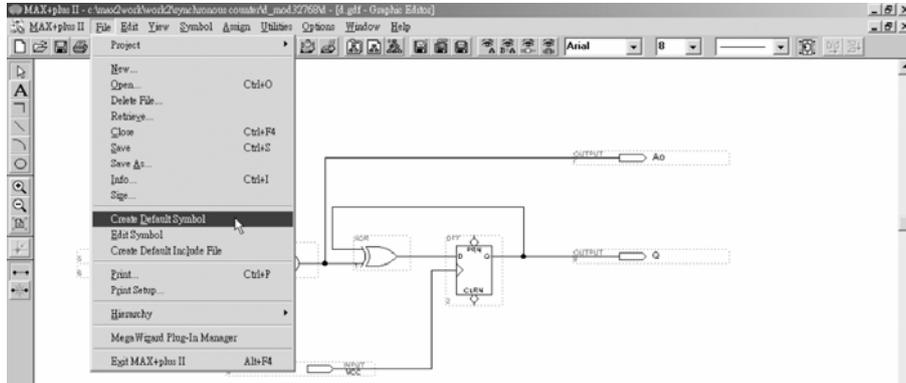
製造計數器元件(symbol)：

1. 開啟一個新圖形編輯檔，繪製一個計數器元件的電路圖，即第二個正反器輸出端之後的邏輯組合，由一個及閘加上一個互斥或閘、再加上一個正反器所組成。
2. 存檔，取檔名為 d.gdf。
3. 編譯。



● 圖 4-3-28 計數器元件電路圖

4. 產生符號檔，點選 **File→Create Default Symbol**，即可產生新的元件(名稱 d.sym)



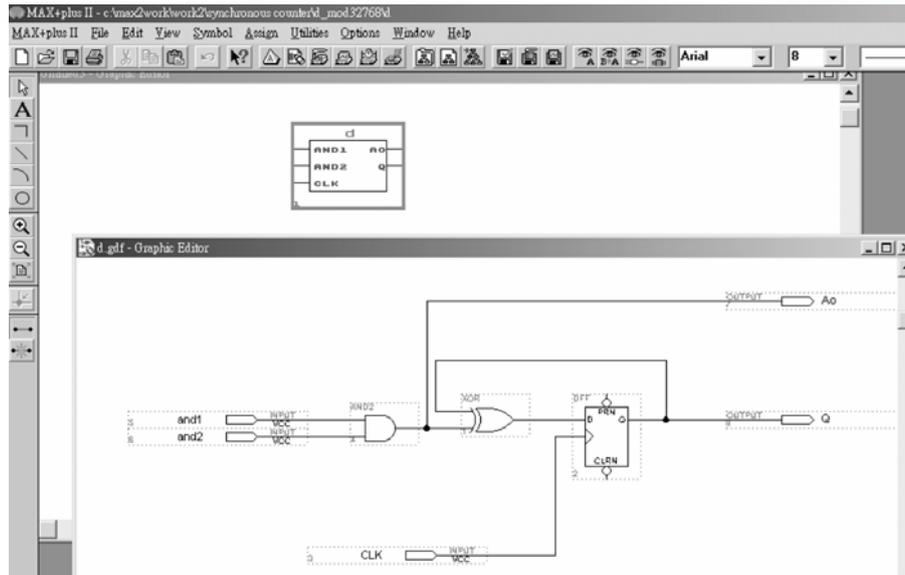
● 圖 4-3-29 執行製作自製元件視窗

5. 開啟新的圖形編輯檔，此時即可取用所製作的元件(d.sym)，點選 **Symbol→Enter Symbol**(或在編輯視窗按兩下 **Double Click**)，出現零件取用視窗。



● 圖 4-3-30 元件取用視窗

6. 取用元件(symbol)：進入剛才存放計數器元件的資料夾，點選元件檔(**Symbol File**) d，按 **OK**，在圖形編輯視窗中，即可看到該元件(包裝後的計數器)。可在計數器元件上按兩下，即可開啟其內部電路視窗。

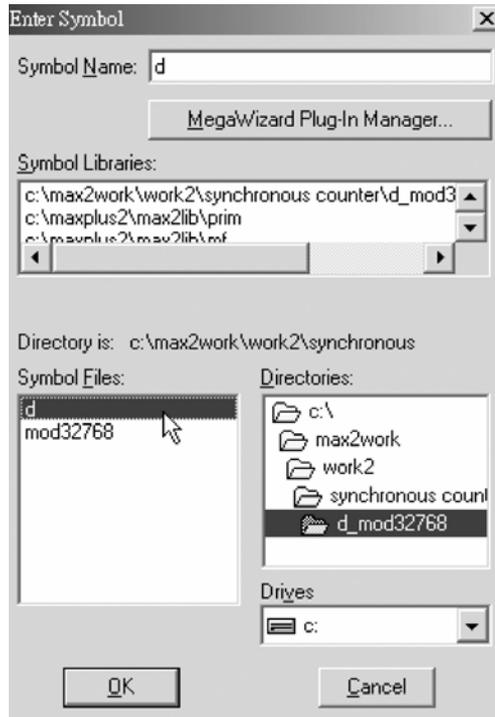


● 圖 4-3-31 計數器自製元件及其內部電路圖

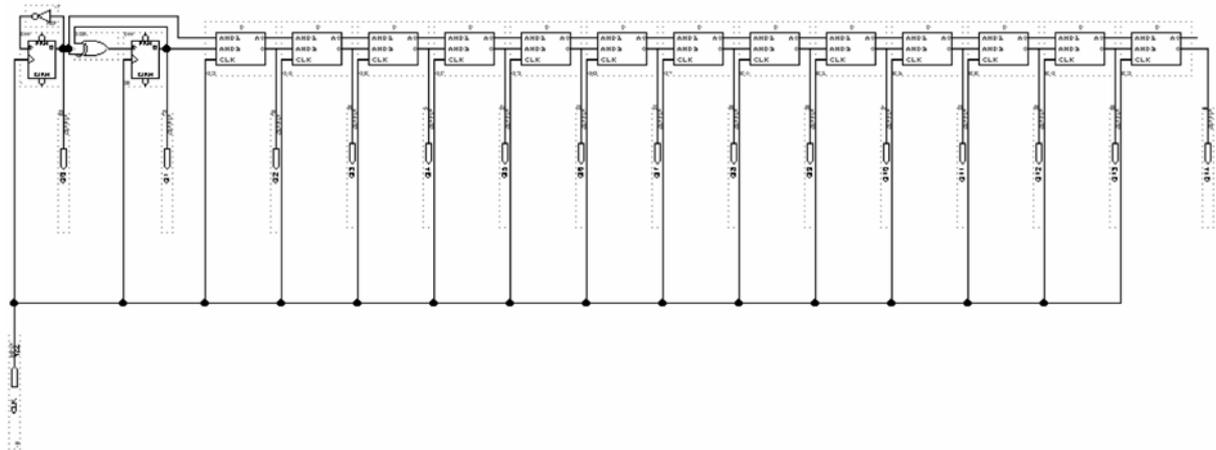
要以自製的計數器元件(symbol)製作 MOD 32768 計數器元件時，步驟如下：

1. 建立一個新資料夾，取名 d\_mod 32768，複製事先製妥計數器元件的圖形檔 (d.gdf)以及元件檔(d.sym)至該資料夾。
2. 執行 MAX+PLUS II 軟體，開啟新的圖形編輯檔，叫出儲存 d\_mod 32768 資料夾的計數器元件檔 d.sym，利用它配合計數器前面電路完成 MOD 32768 計數器繪圖，存檔於 d\_mod 32768 資料夾內。如圖 4-3-33 所示，在此取名 MOD<sub>32768</sub> 計數器圖形檔案名稱爲 mod 32768.gdf。

(因為電路圖甚大，讀者若在書本上看不清楚，可在隨書附贈光碟的 d\_mod 32768 資料夾內找到此檔。)



● 圖 4-3-32 元件取用視窗

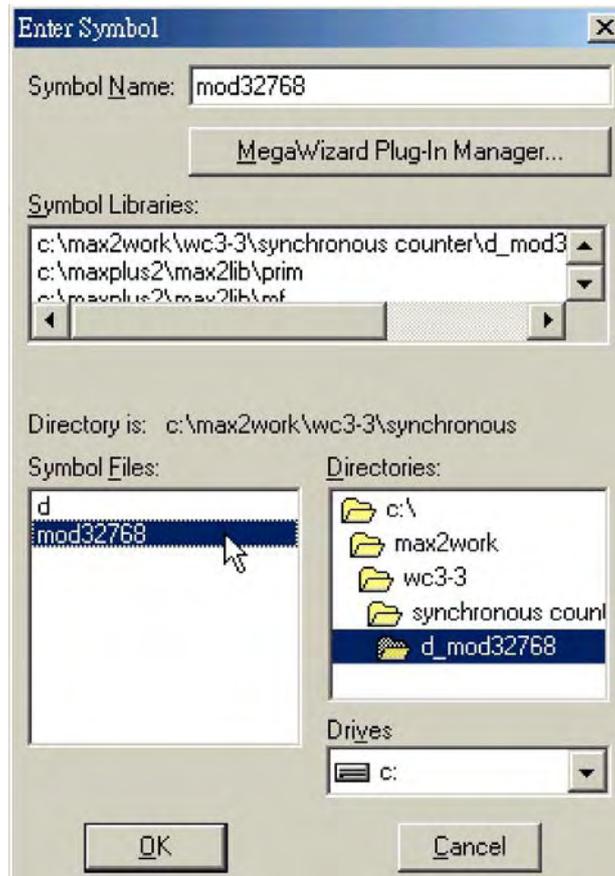


● 圖 4-3-33 MOD 32768 計數器電路圖

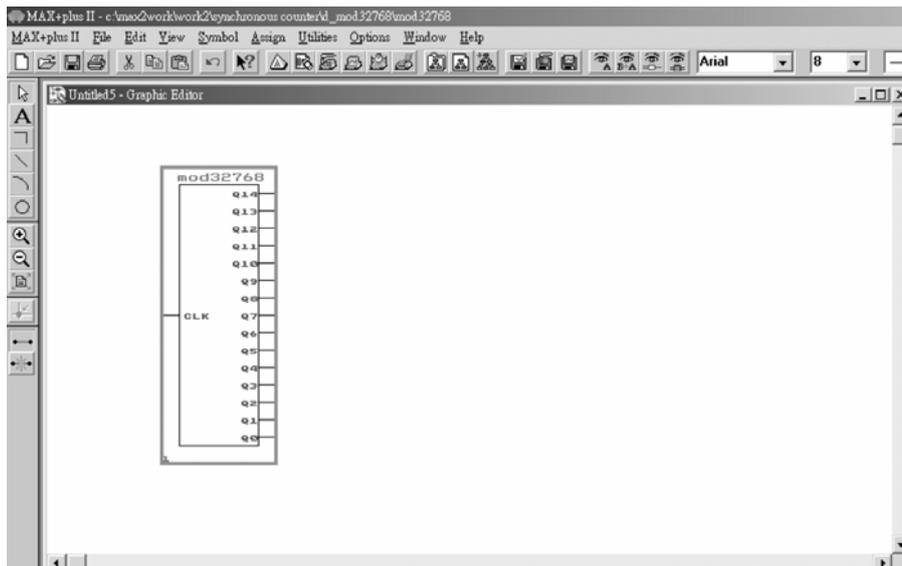


● 圖 4-3-34 儲存檔案視窗

3. [File→Project→Set Project to Current File](#)。
4. 編譯。
5. 產生符號檔，點選 [File→Create Default Symbol](#)，即可產生新的元件(symbol) (MOD32768 計數器，名稱 mod 32768)(檔名：mod 32768.sym)。
6. 開啟新的圖形編輯檔，此時即可取用所製作的元件(mod32768)，點選 [Symbol → Enter Symbol](#)，按確定後，即出現元件取用視窗，選擇剛才所建立之新資料夾 d\_mod32768 即可發現已產生一個新的元件檔 mod32768.sym，選擇此元件即可在圖形編輯視窗中使用它。



● 圖 4-3-35 元件取用視窗



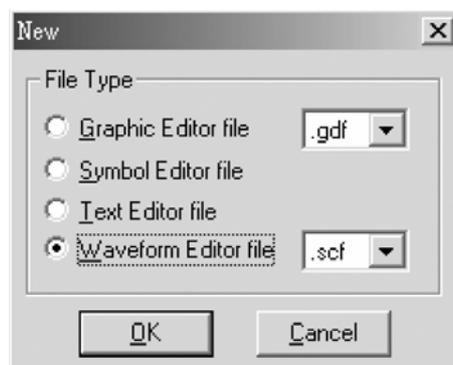
● 圖 4-3-36 自製元件 MOD 32768 計數器

設計 MOD 32768 計數器完成後，要知道其是否可正常執行，可執行模擬功能測試。

模擬步驟如下：

1. 在開啟編輯好的 MOD32768 計數器 d\_mod32768.gdf 檔案，設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。
2. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定 65.536ms，設定格線間距(Options → Grid Size)設定 1  $\mu$ s，顯示在視窗中適當大小格線(View→Fit in Window)。

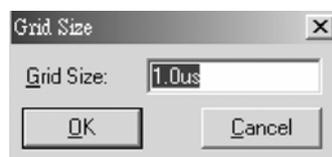
(因為在此輸入端有 1 支腳，輸出端有 15 支腳，共有 16 支腳，每支腳有兩種狀態(0 與 1)，總共有 65536 種變化。



● 圖 4-3-37 開啟新檔視窗

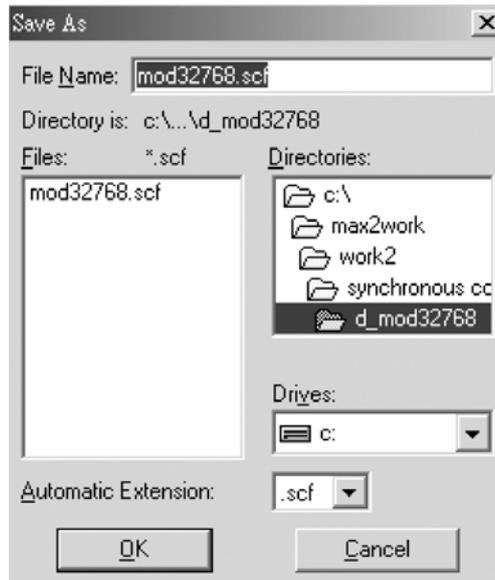


● 圖 4-3-38 模擬結束時間設定視窗

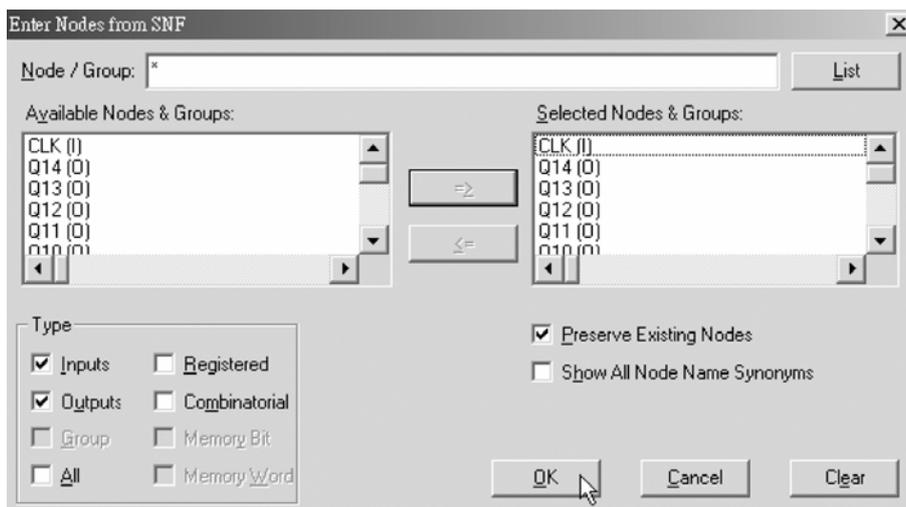


● 圖 4-3-39 模擬單位時間設定視窗

3. 儲存檔案(Save As)，檔名 mod32768.scf，輸入節點(Node → Enter Nodes from SNF，按 List 及  $\geq$ ，OK)，編輯輸入信號，共有 65536  $\mu$ s，剛好可計數 0 至 32767，可用計數時脈完成輸入信號編輯。



● 圖 4-3-40 儲存檔案視窗

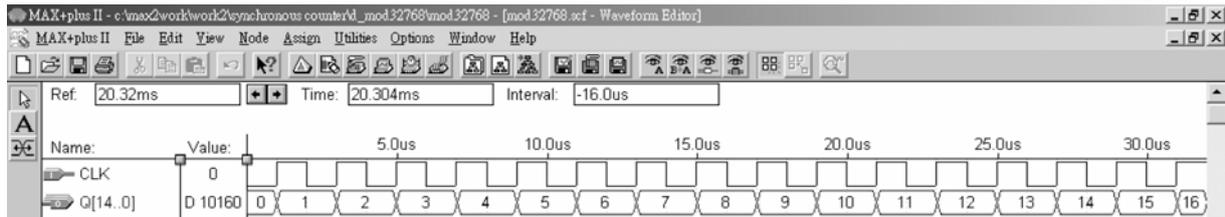


● 圖 4-3-41 輸出入節點選擇視窗

4. 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-3-41 所示，可得各輸出端波形圖。模擬沒有錯誤及警告，所得波形模擬結果符合 MOD32768 計數器，代表我們製作的電路是正確可用的。



● 圖 4-3-42 模擬結果視窗



● 圖 4-3-43 模擬結果視窗

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

● 表 4-3-10 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出							
名稱	CLK	Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76	PIN 77	PIN 79	PIN 80	PIN 81
實驗器模組對應腳位	CLK1	DG0	DG1	DG2	DG3	DG4	DG5	DG6	DG7
腳位對應關係	輸出								
名稱	Q8	Q9	Q10	Q11	Q12	Q13	Q14		
CPLD 晶片腳位	PIN 61	PIN 63	PIN 64	PIN 65	PIN 67	PIN 68	PIN 69		
實驗器模組對應腳位	DR0	DR1	DR2	DR3	DR4	DR5	DR6		

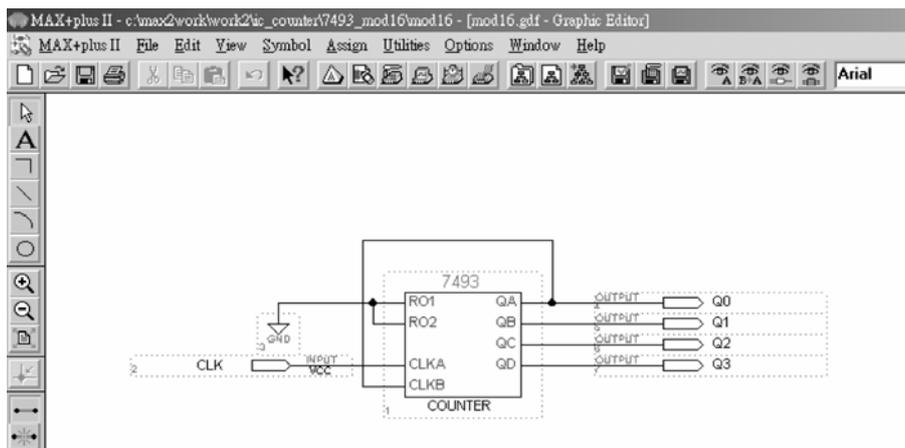
實驗器可選擇使用 CLK<sub>1</sub> 輸入端，選擇由石英晶體振盪輸入時脈 32768Hz。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0~7 個綠色 LED 接腳當輸出端 Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>, Q<sub>4</sub>, Q<sub>5</sub>, Q<sub>6</sub>, Q<sub>7</sub>。第 0~6 個紅色 LED 接腳當輸出端 Q<sub>8</sub>, Q<sub>9</sub>, Q<sub>10</sub>, Q<sub>11</sub>, Q<sub>12</sub>, Q<sub>13</sub>, Q<sub>14</sub>。其中第 7 個紅色 LED DR<sub>6</sub> 接腳 Q<sub>14</sub> 之輸出將為 1Hz。

## 4-4 利用 IC 模組製作計數器

製作計數器時，除了使用正反器可達成外，也可利用計數器 IC 模組來達成，現有的計數器 IC 有很多種，如 TTL 7493 的 MOD16 計數器、7492 的 MOD12 計數器、7490 的 MOD10 BCD 計數器，利用 IC 配合邏輯閘可組成各種計數器。在 MAX+PLUS II 裡，可以直接叫出這些 IC 模組，將它編輯成計數器同樣可下載到 CPLD 中使用，相當方便，以下將介紹作法。

### 4-4-1 7493 MOD16 計數器

利用 TTL 7493 計數器 IC 來製作 MOD 16 計數器時，按照實際硬體電路製作之方法設計線路，7493 IC 有一 MOD 2 及 MOD 8 之計數器，將 MOD 2 計數器輸出送到 MOD8 計數器之輸入，可得 MOD 16 計數器。清除控制腳(R00 及 R01)需接地才不會使其產生清除歸零的動作，設計好後繪製電路圖，即可完成計數器製作。

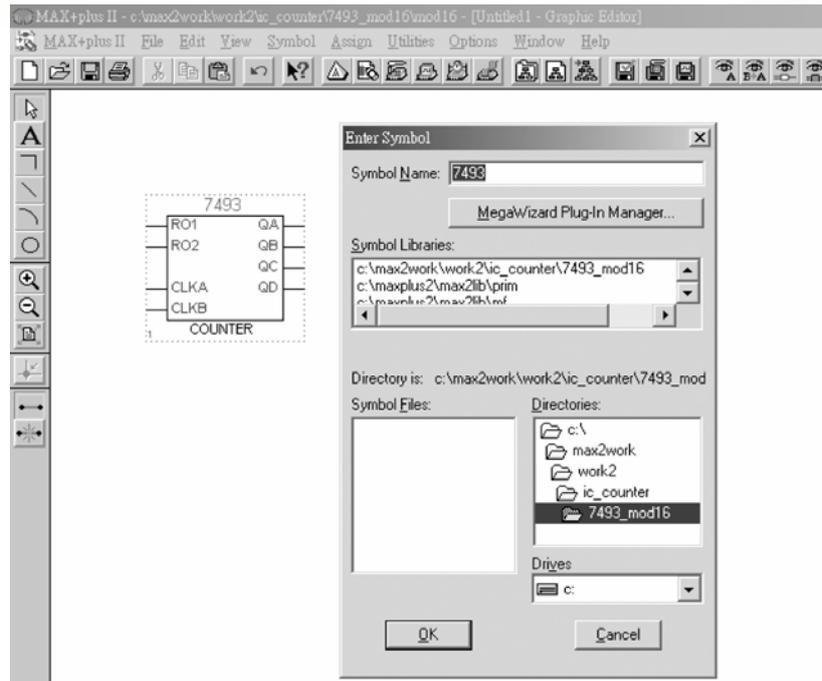


● 圖 4-4-1 7493 MOD 16 計數器電路圖

如圖 4-4-1 所示，經由步驟：**畫電路圖**→**存檔**→**設為工作專案**→**指定 CPLD 晶片**→**編譯**→**軟體模擬**→**規劃腳位**→**編譯**→**下載燒錄**(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

製作 7493MOD 16 計數器步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖，如圖 4-4-2 所示，利用 **Symbol**→**Enter Symbol** 叫出 **Enter Symbol** 視窗，輸入 7493 以叫出計數器 IC 7493 模組，按照設計繪製電路圖。



● 圖 4-4-2 元件取用視窗

2. 新建 7493\_mod16 資料夾後，存檔，取檔名為 mod16.gdf
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

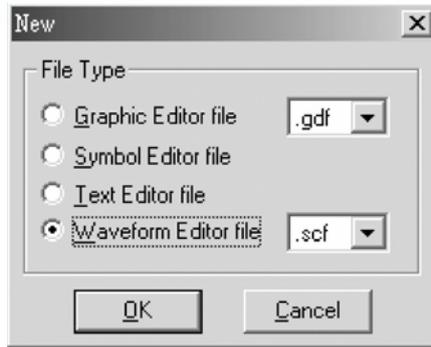
如圖所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



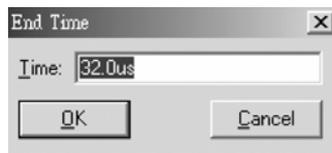
● 圖 4-4-3 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $32 \mu s$ ，設定格線間距(Options→Grid Size)設定  $1 \mu s$ ，顯示在視窗中適當大小格線(View→Fit in Window)。

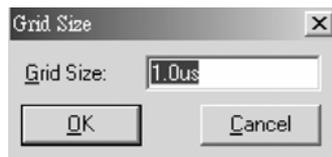
(因為在此計數器輸出有 4 個，可計數 0~15，總共有  $2^4=16$  種變化，當我們設 Grid= $1 \mu s$  時，一個時脈信號需要高態及低態各  $1 \mu s$ ，共需  $2 \mu s$ ，故計數 0~15 需 16 個時脈，總共需  $2 \mu s \times 16 = 32 \mu s$  才完成模擬。)



● 圖 4-4-4 開啟新檔視窗

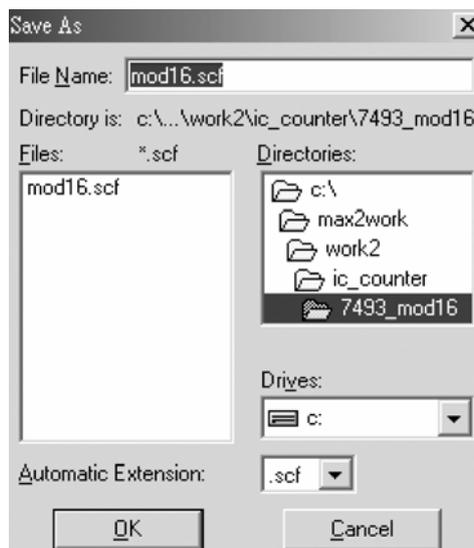


● 圖 4-4-5 模擬結束時間設定視窗

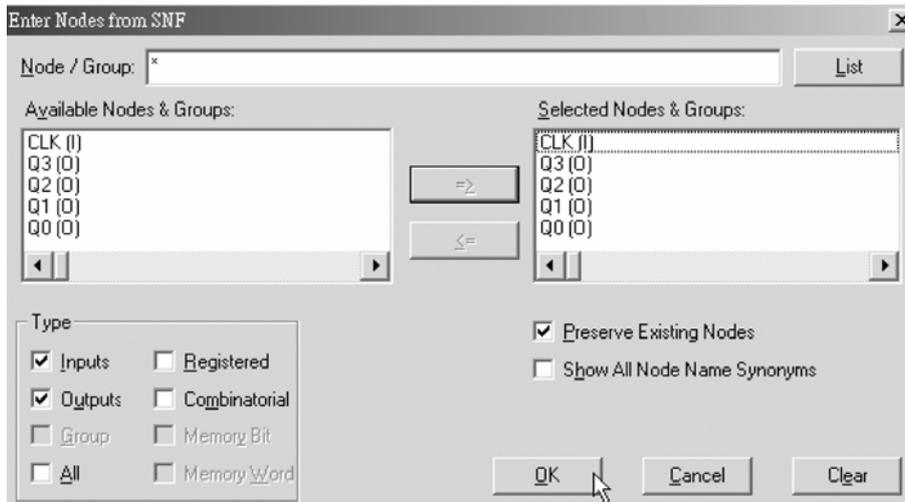


● 圖 4-4-6 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 mod16up.scf，輸入節點(Node→Enter Nodes from SNF，按 List 及 =>，OK)，編輯輸入信號，總共有三十二種組合，剛好 32  $\mu$ s 可模擬完成，可用計數時脈完成輸入信號編輯。

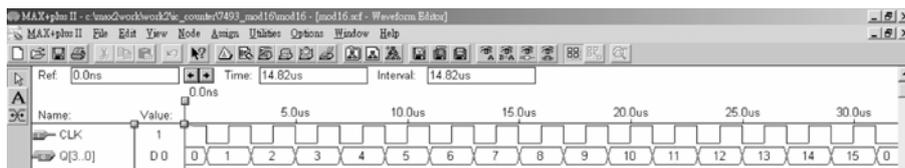


● 圖 4-4-7 儲存檔案視窗

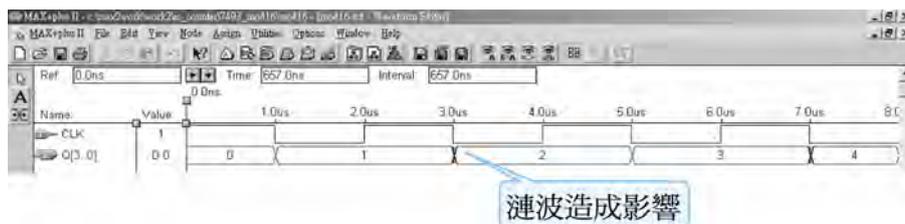


● 圖 4-4-8 輸出入節點選擇視窗

6. 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-4-9 所示，可點選 Q[3..0]，按滑鼠右鍵選擇 Ungroup，可分離得各輸出端波形圖。模擬沒有錯誤及警告，所得波形模擬結果符合 MOD16 計數器，代表我們製作的電路是正確可用的。



● 圖 4-4-9 模擬結果視窗



● 圖 4-4-10 模擬結果視窗

補充一點：利用   調整大小，與上一單元的非同步計數器相同情況，在有很多時間點有出現漣波未定態(如  $3\mu s$  時的上下振盪)，這是因為在 7493 IC 電路設計是前面計數器輸出推動後面計數器輸入的漣波計數方式，因此其會有不同計數器產生的延遲觸發間距的缺點，這電路是屬非同步計數器。

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

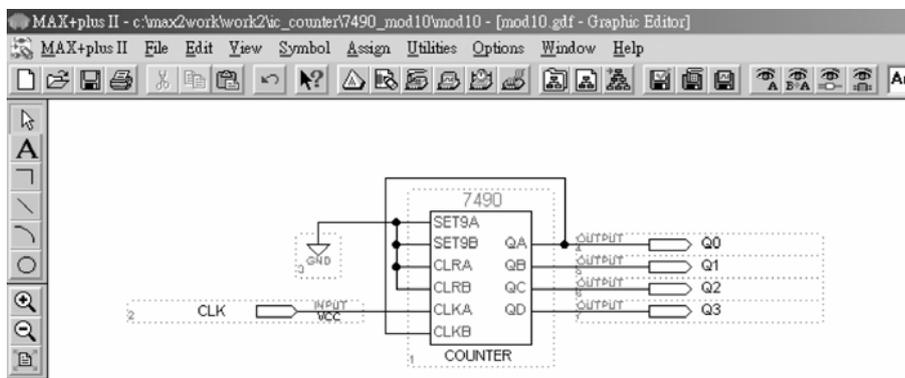
● 表 4-4-1 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出			
名稱	CLK	Q0	Q1	Q2	Q3
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76
實驗器模組對應腳位	CLK <sub>1</sub>	DG0	DG1	DG2	DG3

實驗器可選擇使用 CLK<sub>1</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR<sub>1</sub> 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3 個綠色 LED 接腳當輸出端 Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>。

#### 4-4-2 7490 MOD10 計數器

利用 TTL 7490 計數器 IC 來製作 MOD 10 計數器時，直接按照實際硬體電路製作之方法設計線路，7490IC 有一 MOD 2 及 MOD 5 之計數器，將 MOD 2 計數器輸出送到 MOD 5 計數器之輸入，可得 MOD 10 計數器。清除控制腳(CLR<sub>A</sub> 及 CLR<sub>B</sub>)以及設定控制腳(SET 9A 及 SET 9B)需接地才不會使其產生清除歸零或設定為 1 的動作，設計好後繪製電路圖，即可完成計數器製作。



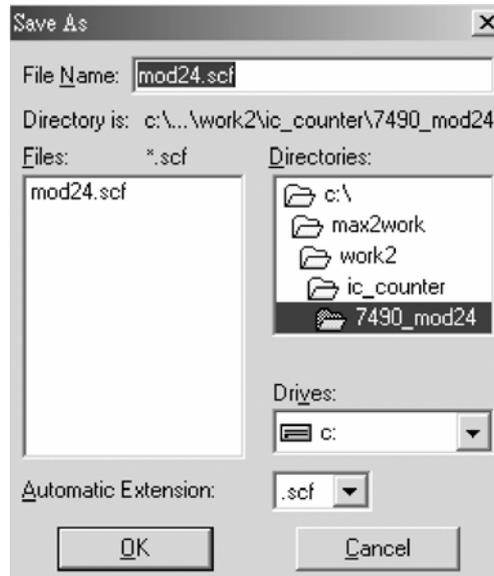
● 圖 4-4-11 7490 MOD 10 計數器電路圖

如圖 4-4-11 所示，經由步驟：**畫電路圖**→**存檔**→**設為工作專案**→**指定 CPLD 晶片**→**編譯**→**軟體模擬**→**規劃腳位**→**編譯**→**下載燒錄**(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

製作 7490MOD 10 計數器步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖，如圖 4-4-12 所示，利用 **Symbol**→**Enter Symbol** 叫出 **Enter Symbol** 視窗，輸入 7490 以叫出計數器 IC 7490 模組，按

照設計繪製電路圖。



● 圖 4-4-12 元件取用視窗

2. 新建 7490\_mod10 資料夾後，存檔，取檔名為 mod10.gdf。
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

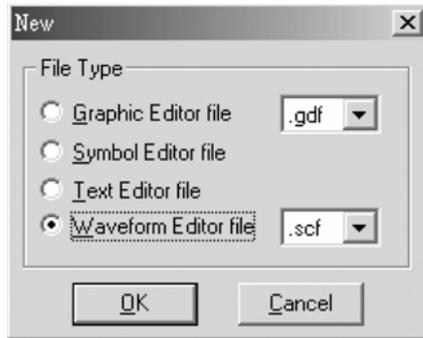
如圖 4-4-13 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



● 圖 4-4-13 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $20 \mu s$ ，設定格線間距(Options→Grid Size)設定  $1 \mu s$ ，顯示在視窗中適當大小格線(View→Fit in Window)。

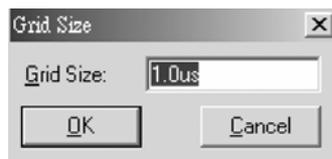
(因為在此為 MOD10 計數器，每個計數脈波有兩種狀態(0 與 1)，總共有 20 種變化。)



● 圖 4-4-14 開啟新檔視窗

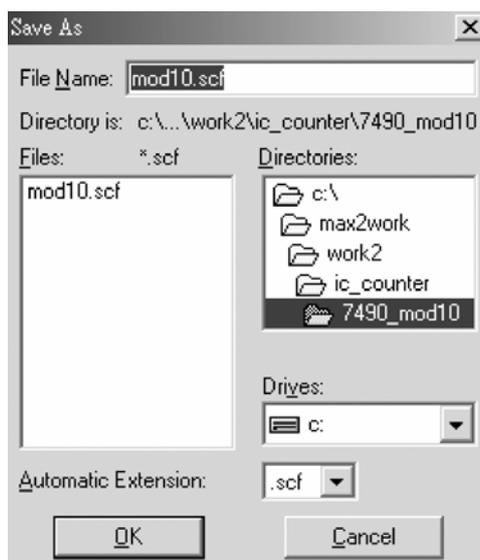


● 圖 4-4-15 模擬結束時間設定視窗

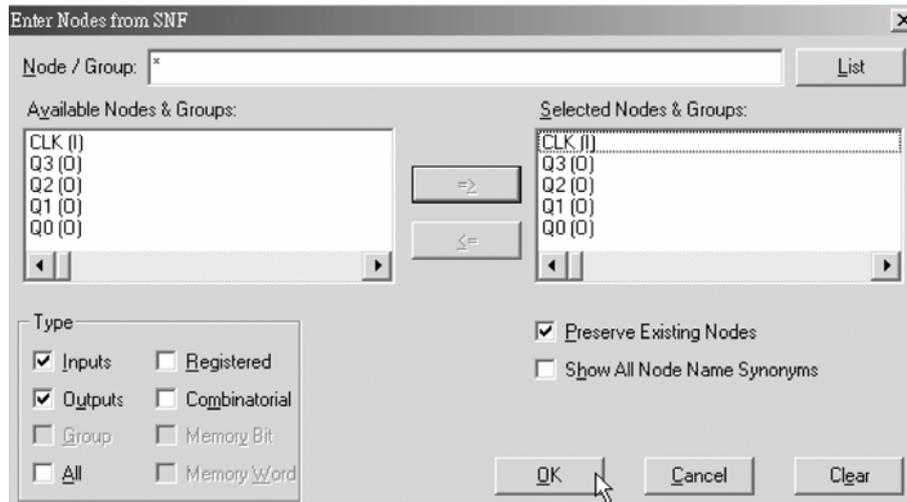


● 圖 4-4-16 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 `mod10.scf`，輸入節點(Node→Enter Nodes from SNF，按 List 及 ⇒，OK)，編輯輸入信號，計數 0~9 十種狀態，剛好  $20\mu s$  可模擬完成，可用計數時脈完成輸入信號編輯。



● 圖 4-4-17 儲存檔案視窗



● 圖 4-4-18 輸出入節點選擇視窗

6. 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-4-19 所示,可點選 **Q[3..0]**, 按滑鼠右鍵選擇 **Ungroup**, 可得各輸出端波形圖。模擬沒有錯誤及警告, 所得波形模擬結果符合 MOD10 計數器, 代表我們製作的電路是正確可用的。



● 圖 4-4-19 模擬結果視窗

補充一點：與上一單元的非同步計數器相同情況，在有很多時間點有出現漣波未定態(如  $3\mu s$  時的上下振盪)，這是因為在 7490 IC 與 7493 是相同的方式，是前面計數器輸出推動後面計數器輸入的漣波計數方式，因此其會有不同計數器產生的延遲觸發間距的缺點，這電路亦是屬非同步計數器。

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

● 表 4-4-2 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出			
		Q0	Q1	Q2	Q3
名稱	CLK				
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76
實驗器模組對應腳位	CLK1 或 CLK2	DG0	DG1	DG2	DG3

實驗器可選擇使用 CLK<sub>1</sub> 或 CLK<sub>2</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR<sub>1</sub> 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3 個綠色 LED 接腳當輸出端 Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, Q<sub>3</sub>。

#### 4-4-3 7490 MOD 6 計數器

TTL 7490 計數器 IC 除了可製作 MOD10 計數器，也可製作 MOD10 以下的各種計數器，以下介紹使用其控制腳設定(SET)及清除(CLEAR)來製作 MOD6 計數器。設計技巧與實際硬體電路製作之方法一樣，7490 IC 有一 MOD 2 及 MOD 5 之計數器，將 MOD 2 計數器輸出送到 MOD 5 計數器之輸入，可得 MOD 10 計數器。在其計數 Q<sub>D</sub>Q<sub>C</sub>Q<sub>B</sub>Q<sub>A</sub> 為 0000~0101 (0~5)時為正常計數，當計數值數到 0110 (6，第 7 種狀態)時，則給予清除控制腳(CLR<sub>A</sub> 及 CLR<sub>B</sub>)接收 1 信號，達到清除使輸出 Q<sub>D</sub>Q<sub>C</sub>Q<sub>B</sub>Q<sub>A</sub> 為 0000。設定控制腳(SET<sub>9A</sub> 及 SET<sub>9B</sub>)需接地才不會使其產生設定為 1 的動作，如此便完成設計好後繪製電路圖，即可完成 MOD 6 計數器製作。

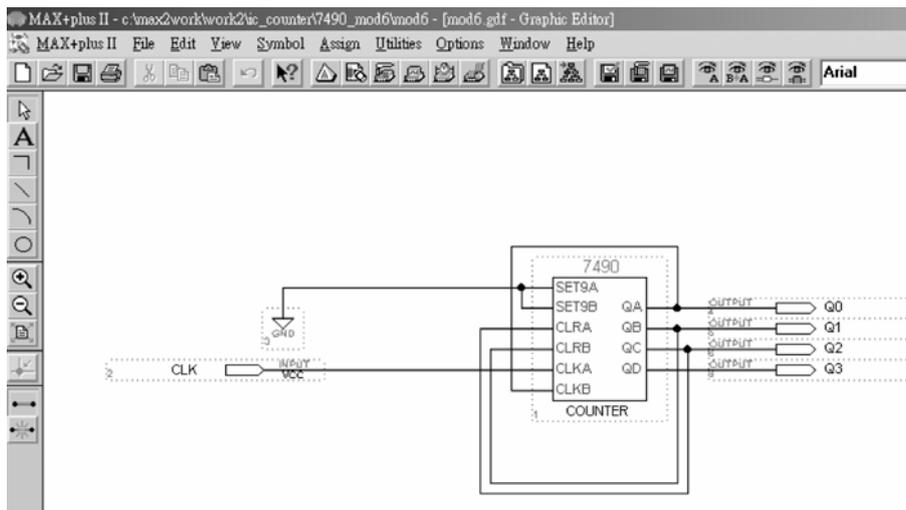


圖 4-4-20 7493 MOD6 計數器電路圖

如圖 4-4-20 所示，經由步驟：**畫電路圖**→**存檔**→**設為工作專案**→**指定 CPLD 晶片**→**編譯**→**軟體模擬**→**規劃腳位**→**編譯**→**下載燒錄**(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

製作 7490MOD 6 計數器步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖，利用 **Symbol**→**Enter Symbol** 叫出 **Enter Symbol** 視窗，輸入 7490 以叫出計數器 IC 7490，按照設計繪製電路圖。

2. 新建 7490\_mod6 資料夾後，存檔，取檔名為 mod6.gdf
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II →Compiler)。

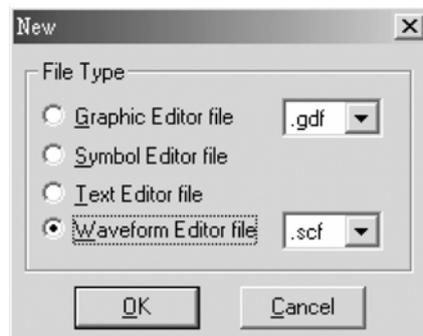
如圖 4-4-21 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



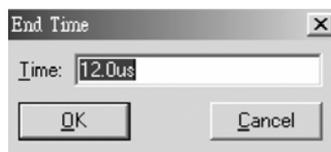
● 圖 4-4-21 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $12\ \mu\text{s}$ ，設定格線間距(Options→Grid Size)設定  $1\ \mu\text{s}$ ，顯示在視窗中適當大小格線(View→Fit in Window)。

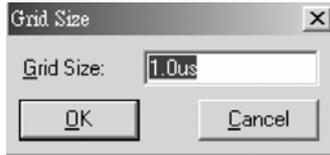
(因為在此為 MOD 6 計數器，每個計數脈波有兩種狀態(0 與 1)，總共有  $6 \times 2 = 12$  種變化。)



● 圖 4-4-22 開啟新檔視窗

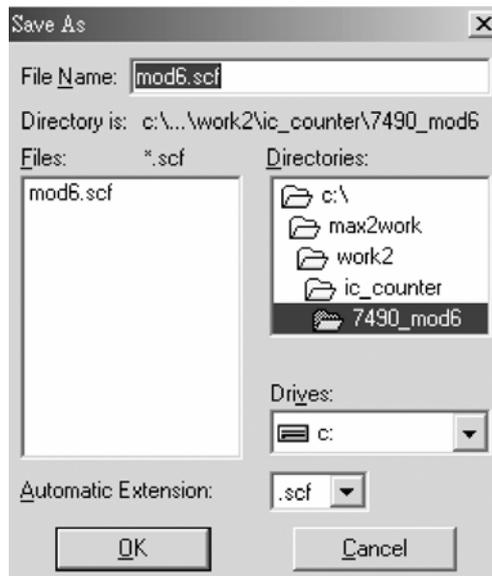


● 圖 4-4-23 模擬結束時間設定視窗

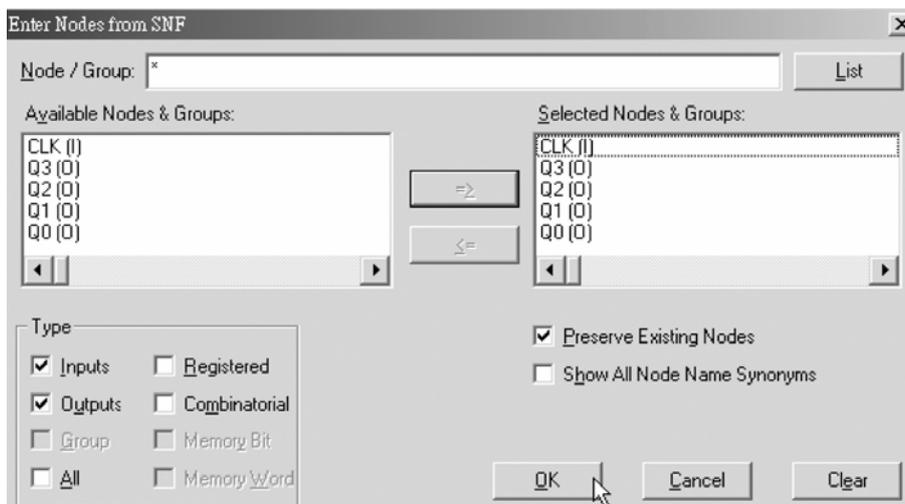


● 圖 4-4-24 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 mod6.scf，輸入節點(Node→Enter Nodes from SNF，按 List 及  $\Rightarrow$ ，OK)，編輯輸入信號，計數 0~5 六種狀態，剛好 12  $\mu$ s 可模擬完成，可用計數時脈完成輸入信號編輯。

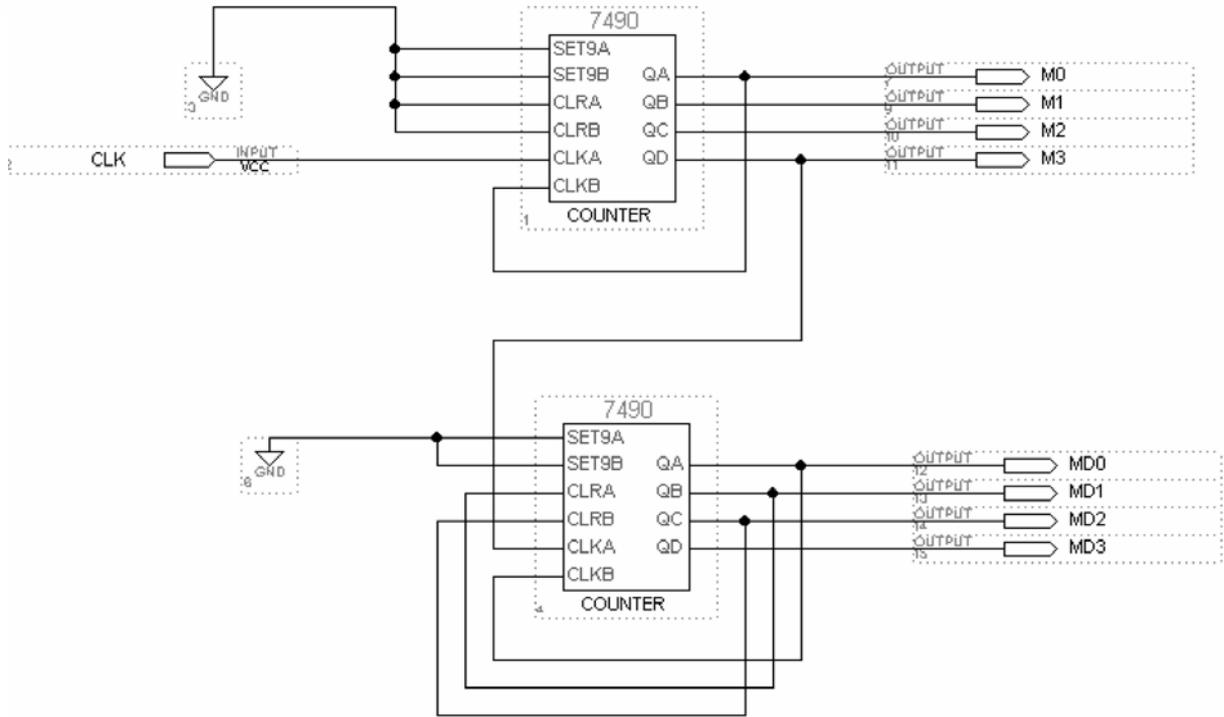


● 圖 4-4-25 儲存檔案視窗



● 圖 4-4-26 輸出入節點選擇視窗





● 圖 4-4-28 7490 MOD60 計數器電路圖

如圖 4-4-28 所示，經由步驟：[畫電路圖](#)→[存檔](#)→[設為工作專案](#)→[指定 CPLD 晶片](#)→[編譯](#)→[軟體模擬](#)→[規劃腳位](#)→[編譯](#)→[下載燒錄](#)(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

製作 7490 MOD 60 計數器步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖，如圖 4-4-28 所示，利用 [Symbol](#)→[Enter Symbol](#) 叫出 [Enter Symbol](#) 視窗，輸入 7490 以叫出計數器 IC 7490，按照設計繪製電路圖。
2. 新建 7490\_mod60 資料夾後，存檔，取檔名為 mod60.gdf
3. 設為指定工作專案([File](#)→[Project](#)→[Set Project to Current File](#))，指定 CPLD 晶片([Assign](#)→[Device](#))，並編譯([MAX+PLUS II](#) → [Compiler](#))。

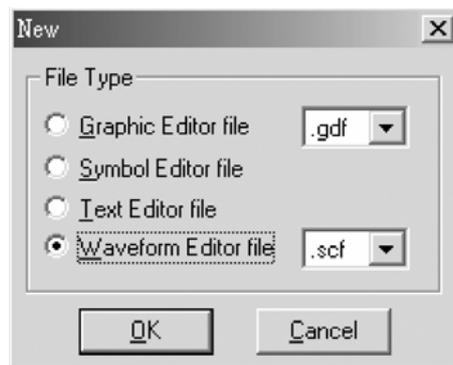
如圖 4-4-29 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



● 圖 4-4-29 編譯訊息視窗

4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File → End Time)設定  $120\ \mu\text{s}$ ，設定格線間距(Options → Grid Size)設定  $1\ \mu\text{s}$ ，顯示在視窗中適當大小格線(View → Fit in Window)。

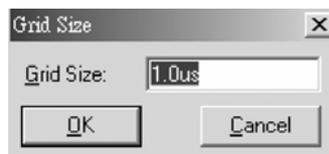
(因為在此為 MOD 60 計數器，每個計數脈波有兩種狀態(0 與 1)，總共有 120 種變化。)



● 圖 4-4-30 開啟新檔視窗



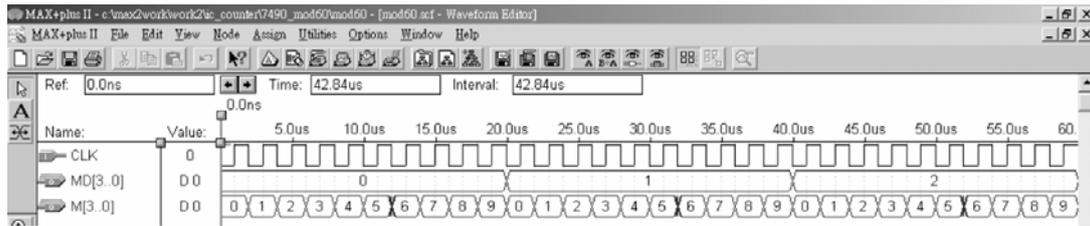
● 圖 4-4-31 模擬結束時間設定視窗



● 圖 4-4-32 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 mod60.scf，輸入節點(Node → Enter Nodes from SNF，按 List 及  $\Rightarrow$ ，OK)，編輯輸入信號，計數 0~59 六十種狀態，剛好  $120\ \mu\text{s}$  可模擬完成，可用計數時脈完成輸入信號編輯。





● 圖 4-4-36 模擬結果視窗

與上一單元的 MOD 6 計數器相同情況，在有很多時間點有出現漣波未定態，此電路亦是屬非同步計數器。

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

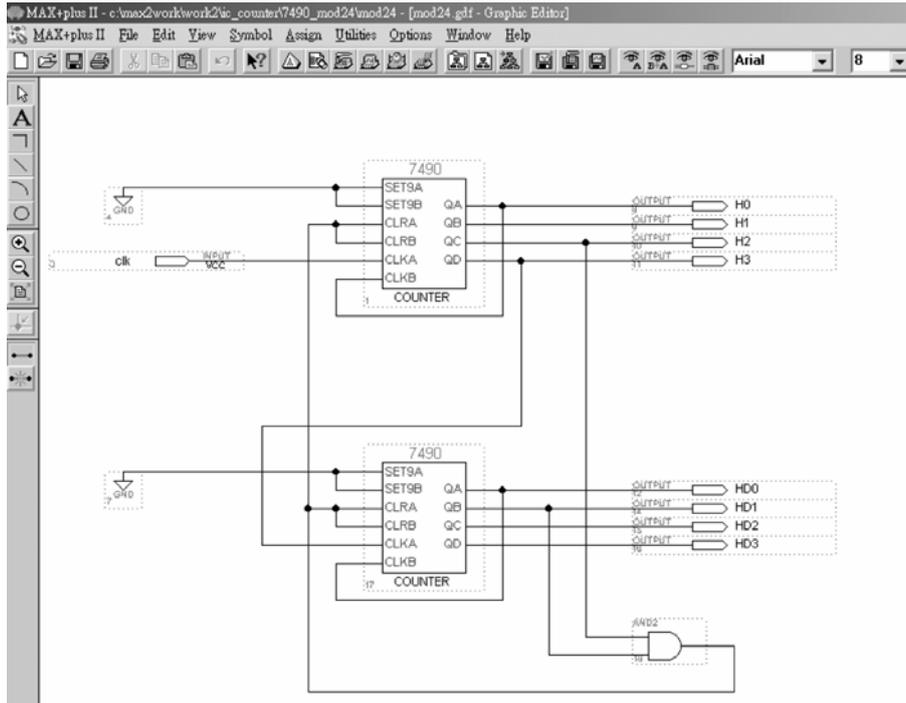
● 表 4-4-4 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出							
名稱	CLK	M0	M1	M2	M3	MD0	MD1	MD2	MD3
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76	PIN 77	PIN 79	PIN 80	PIN 81
實驗器模組對應腳位	CLK1	DG0	DG1	DG2	DG3	DG4	DG5	DG6	DG7

實驗器可選擇使用 CLK<sub>1</sub> 或 CLK<sub>2</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR<sub>1</sub> 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3,4,5,6,7 等 8 個綠色 LED 接腳當輸出端 M<sub>0</sub>；M<sub>1</sub>；M<sub>2</sub>；M<sub>3</sub>；MD<sub>0</sub>；MD<sub>1</sub>；MD<sub>2</sub>；MD<sub>3</sub>。

#### 4-4-5 7490 MOD 24

利用 TTL 7490 計數器 IC 製作 MOD24 計數器，先將兩個 MOD10 計數器電路串級後，製作 MOD100 計數器，再利用邏輯閘使其一計數到 24 時立即使清除(CLEAR)腳為 1，使計數清除歸零。即可達成 0~23 計數之 MOD24 計數器。如圖 4-4-37 所示，H<sub>3</sub>H<sub>2</sub>H<sub>1</sub>H<sub>0</sub> 為低位元的個位數，HD<sub>3</sub> HD<sub>2</sub> HD<sub>1</sub> HD<sub>0</sub> 為高位元的十位數。當計數器計數到 24 時，個位數的 7490 輸出 H<sub>3</sub>H<sub>2</sub>H<sub>1</sub>H<sub>0</sub> 為 0100，十位數的 7490 輸出 HD<sub>3</sub> HD<sub>2</sub> HD<sub>1</sub> HD<sub>0</sub> 為 0010，此時 H<sub>2</sub> 及 HD<sub>1</sub> 經由 AND 閘輸出 1，其連接到兩個 7490 的清除控制端，而使計數器清除為 0，繼續 0~23 之計數。



● 圖 4-4-37 7490 MOD24 計數器電路圖

如圖 4-4-37 所示，經由步驟：**畫電路圖**→**存檔**→**設為工作專案**→**指定 CPLD 晶片**→**編譯**→**軟體模擬**→**規劃腳位**→**編譯**→**下載燒錄**(若是 Atmel 的晶片下載前需轉檔)後，即完成電路設計製作，此時可配合外部電路(或實驗器模組，需配合腳位規劃)來驗證。

製作 7490 MOD 24 計數器步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖，利用 **Symbol**→**Enter Symbol** 叫出 Enter Symbol 視窗，輸入 7490 以叫出計數器 IC 7490，按照設計繪製電路圖。
2. 新建 7490\_mod24 資料夾後，存檔，取檔名為 mod24.gdf
3. 設為指定工作專案(**File**→**Project**→**Set Project to Current File**)，指定 CPLD 晶片(**Assign**→**Device**)，並編譯(**MAX+PLUS II**→**Compiler**)。

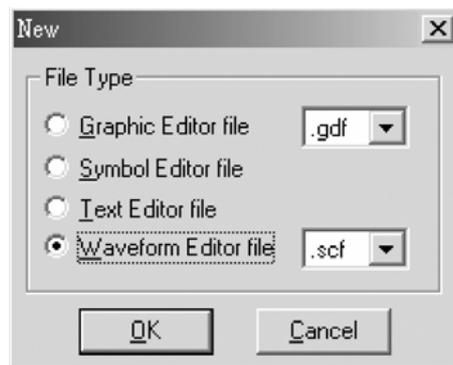
如圖 4-4-38 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



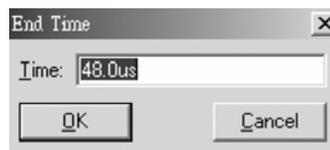
● 圖 4-4-38 編譯訊息視窗

- 開啟新的波形編輯檔案，設定功能模擬結束時間 (File→End Time) 設定 48，設定格線間距 (Options → Grid Size) 設定  $1\ \mu\text{s}$ ，顯示在視窗中適當大小格線 (View→Fit in Window)。

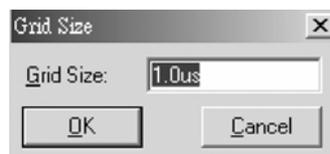
(因為在此為 MOD24 計數器，每個計數脈波有兩種狀態(0 與 1)，總共有 48 變化。)



● 圖 4-4-39 開啟新檔視窗

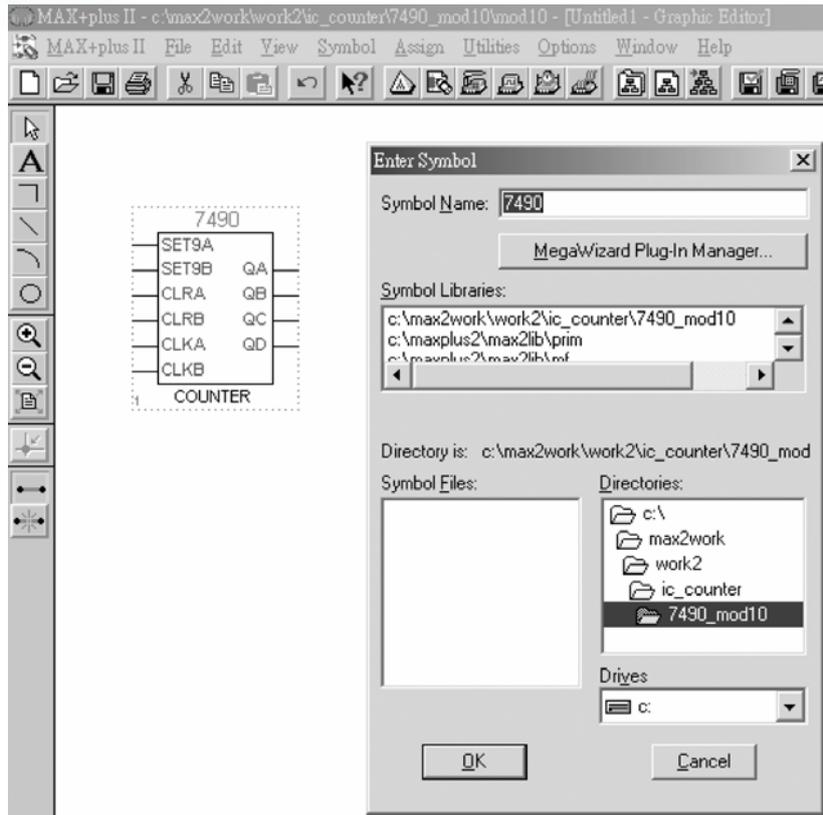


● 圖 4-4-40 模擬結束時間設定視窗

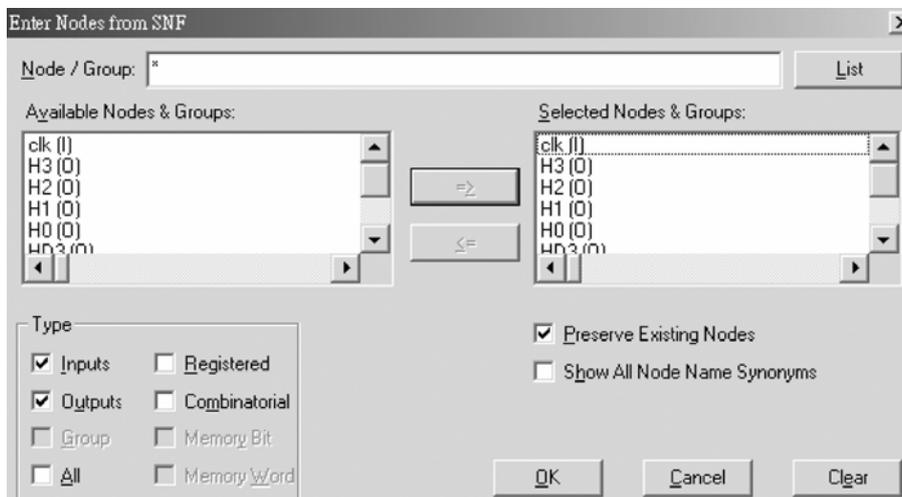


● 圖 4-4-41 模擬單位時間設定視窗

- 儲存檔案(Save As)，檔名 mod60.scf，輸入節點(Node → Enter Nodes from SNF，按 List 及  $\Rightarrow$ ，OK)，編輯輸入信號，計數 0~23 計二十四種狀態，剛好  $48\ \mu\text{s}$  可模擬完成，可用計數時脈完成輸入信號編輯。

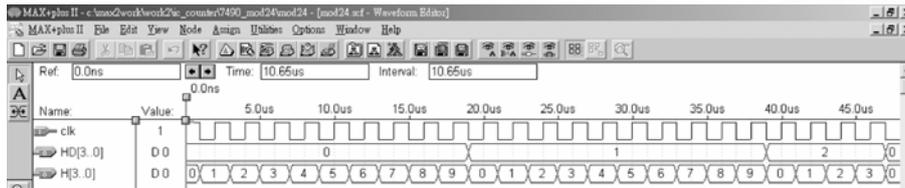


● 圖 4-4-42 儲存檔案視窗

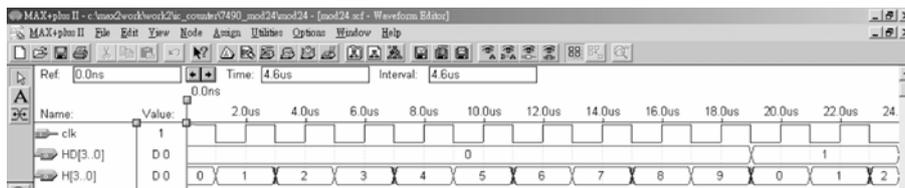


● 圖 4-4-43 輸出入節點選擇視窗

- 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-4-44 所示,可點選 HD[3..0] 及 H[3..0], 按滑鼠右鍵選擇 Ungroup, 可得各輸出端波形圖。模擬沒有錯誤及警告, 所得波形模擬結果符合 MOD 24 計數器, 代表我們製作的電路是正確可用的。



● 圖 4-4-44 模擬結果視窗



● 圖 4-4-45 模擬結果視窗

與上一單元的 MOD 60 計數器相同情況，在有很多時間點有出現漣波未定態，此電路亦是屬非同步計數器。

模擬成功之後，可下載(燒錄)到實驗板做實際電路測試，以下為接到尼德公司實驗板的接腳表格，讀者可以按表中接腳設定以完成硬體電路測試。

● 表 4-4-5 電路圖輸出入腳位、CPLD 腳位及實驗器模組腳位對應表

腳位對應關係	輸入	輸出							
名稱	CLK	H0	H1	H2	H3	HD0	HD1	HD2	HD3
CPLD 晶片腳位	PIN 83	PIN 73	PIN 74	PIN 75	PIN 76	PIN 77	PIN 79	PIN 80	PIN 81
實驗器模組對應腳位	CLK1	DG0	DG1	DG2	DG3	DG4	DG5	DG6	DG7

實驗器可選擇使用 CLK<sub>1</sub> 輸入端，可選擇由石英晶體振盪輸入時脈 32768Hz 或 RC 振盪電路(調整 VR<sub>1</sub> 可改變頻率)。84PIN 之 CPLD 其時脈輸入接腳必須用第 83PIN，而 44PIN 之 CPLD 則需用第 43PIN。在此需以 83PIN 接腳當時脈輸入端，第 0,1,2,3,4,5,6,7 個綠色 LED 接腳當輸出端 H<sub>0</sub>；H<sub>1</sub>；H<sub>2</sub>；H<sub>3</sub>；HD<sub>0</sub>；HD<sub>1</sub>；HD<sub>2</sub>；HD<sub>3</sub> 之輸出。

## 4-5 微分電路

在使用正反器之時，有時會用到前緣觸發及後緣觸發的功能，主要是要觸發時間只在時脈狀態改變之時，而非整個時脈週期都可觸發。因為此電路只在時脈由 1 變 0 或由 0 變 1 時產生一極小週期的觸發信號，類似電路之微分作用，故稱微分電路。

### 4-5-1 前微分電路

如圖 4-5-1 所示為前微分電路之輸出入波形圖，對照圖 4-5-2 電路圖，其原理為只有當輸入信號由 0 變 1 之時，第一個 D 型正反器輸出 1，第二個正反器仍保留上一個狀態 0 輸出之時，輸出端 f\_diff 才會輸出 1，而且只維持 1 個時脈週期，當下一個時脈週期時，兩個正反器輸出皆為 1，此時 f\_diff 又回復 0 之狀態，在此可藉由時脈週期(CLOCK)來調整輸出 f\_diff 之高態時間長短。

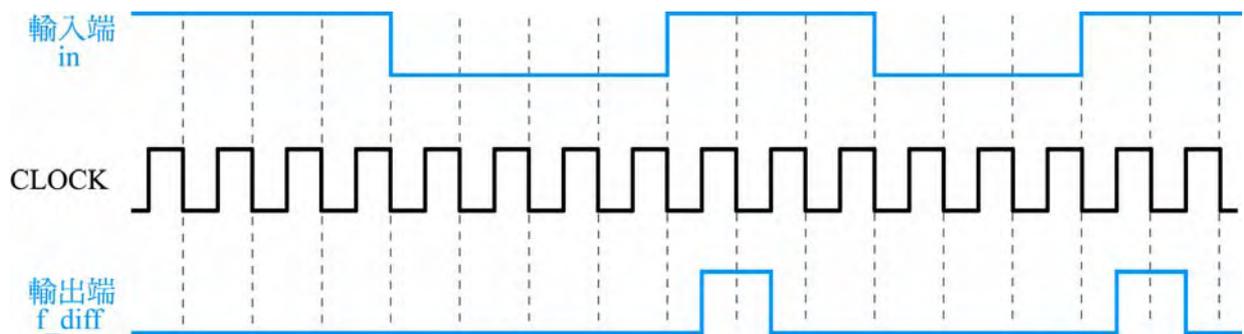


圖 4-5-1 前微分電路波形示意圖

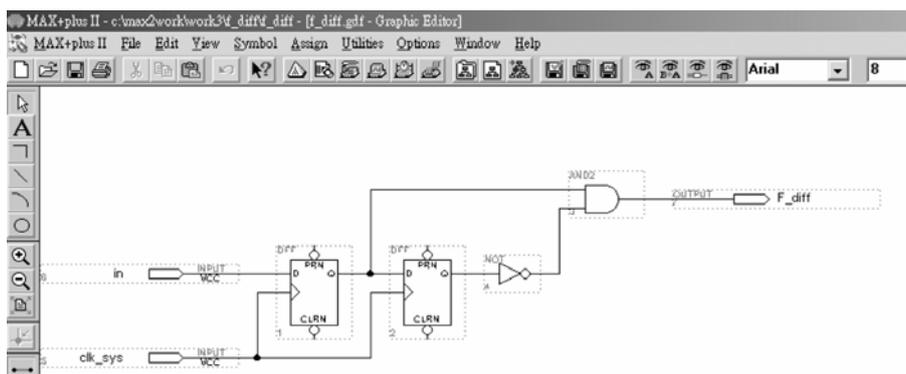


圖 4-5-2 前微分電路圖

製作前微分電路步驟如下：

1. 開啟新圖形編輯檔，繪製邏輯電路圖。
2. 新建 f\_diff6 資料夾後，存檔，取檔名為 f\_diff.gdf。

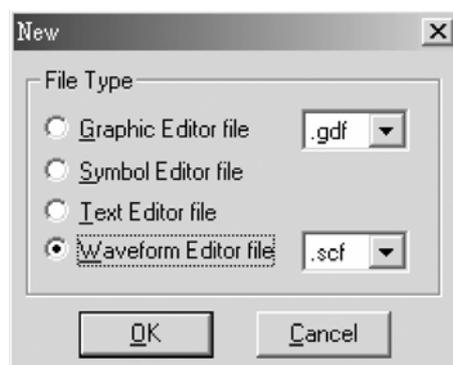
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

如圖 4-5-3 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。

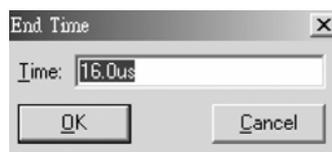


● 圖 4-5-3 編譯訊息視窗

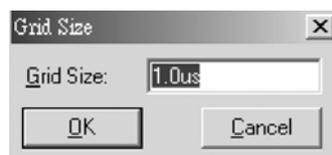
4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $16 \mu s$ ，設定格線間距(Options→Grid Size)設定  $1 \mu s$ ，顯示在視窗中適當大小格線(View→Fit in Window)。



● 圖 4-5-4 開啟新檔視窗



● 圖 4-5-5 模擬結束時間設定視窗



● 圖 4-5-6 模擬單位時間設定視窗

- 5. 儲存檔案(Save As)，檔名 f\_diff.scf，輸入節點(Node → Enter Nodes from SNF，按 List 及 ⇒，OK)，編輯輸入信號。

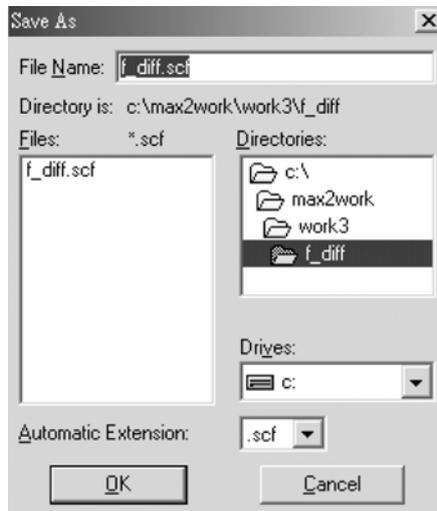


圖 4-5-7 儲存檔案視窗

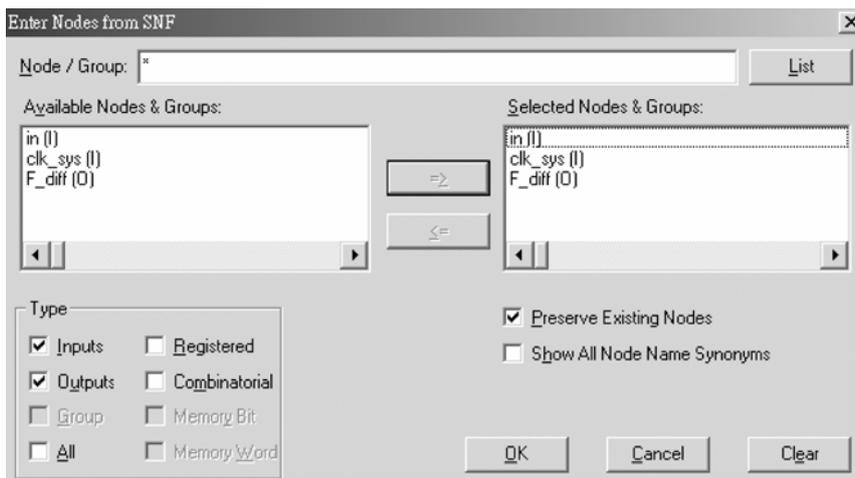


圖 4-5-8 輸出入節點選擇視窗

- 6. 執行模擬(MAX+PLUS II → Simulator，Start)如圖 4-5-9 所示，模擬沒有錯誤及警告，所得波形模擬結果符合前微分電路，代表我們製作的電路是正確可用的。

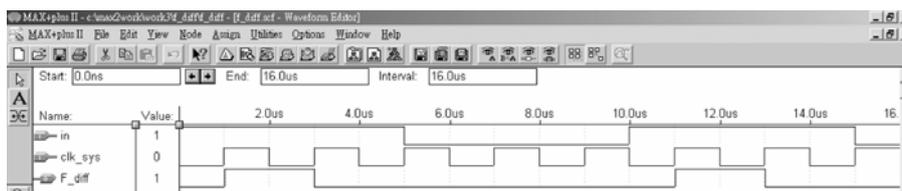
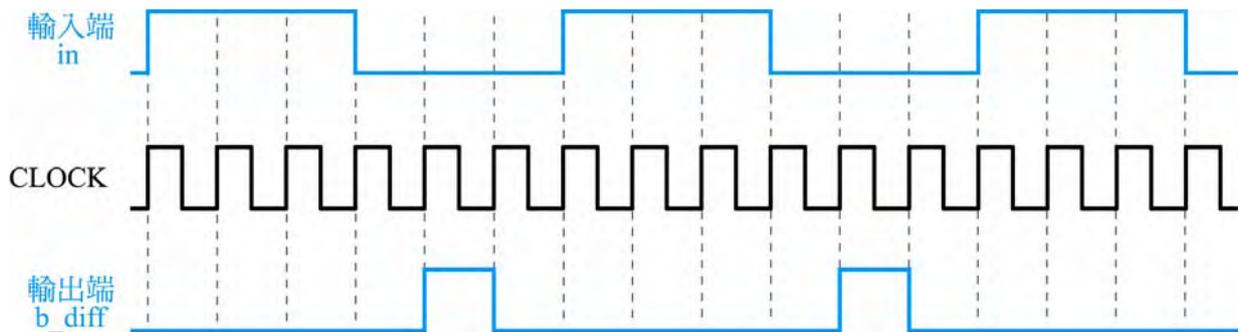


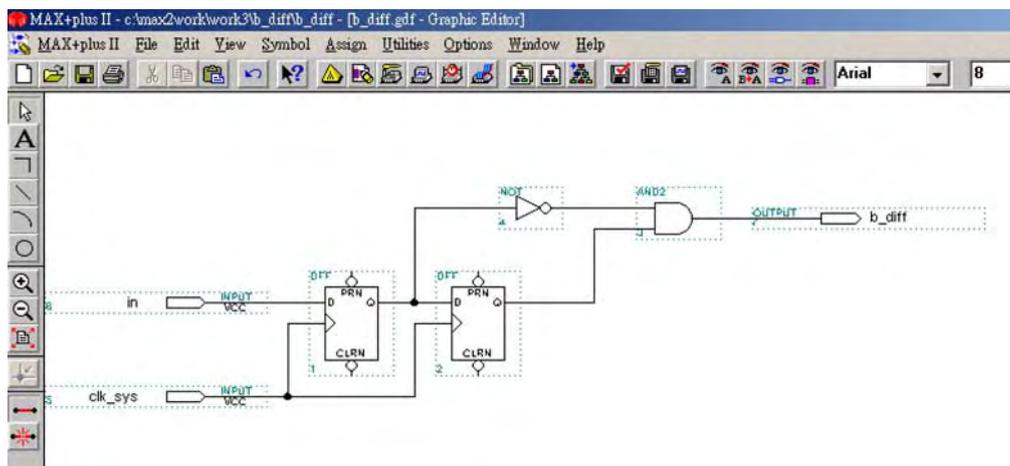
圖 4-5-9 模擬結果視窗

### 4-5-2 後微分電路

如圖 4-5-10 所示為後微分電路之輸出入波形圖，對照圖 4-5-11 電路圖，其原理為只有當輸入信號由 1 變 0 之時，第一個 D 型正反器輸出 0，第二個正反器仍保留上一個狀態 1 輸出之時，輸出端 b\_diff 才會輸出 1，而且只維持 1 個時脈週期，當下一個時脈週期時，兩個正反器輸出皆為 0，此時 b\_diff 又回復 0 之狀態，在此可藉由時脈週期(CLOCK)來調整輸出 b\_diff 之高態時間長短。



● 圖 4-5-10 後微分電路波形示意圖



● 圖 4-5-11 後微分電路圖

製作後微分電路步驟如下：

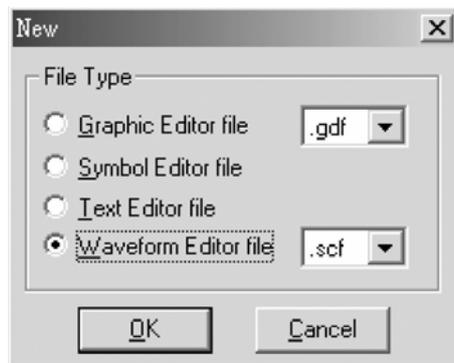
1. 開啟新圖形編輯檔，繪製邏輯電路圖。
2. 新建 b\_diff6 資料夾後，存檔，取檔名為 b\_diff.gdf。
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

如圖 4-5-12 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



● 圖 4-5-12 編譯訊息視窗

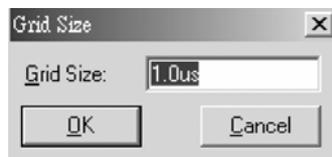
4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $16\ \mu s$ ，設定格線間距(Options→Grid Size)設定  $1\ \mu s$ ，顯示在視窗中適當大小格線(View→Fit in Window)。



● 圖 4-5-13 開啟新檔視窗

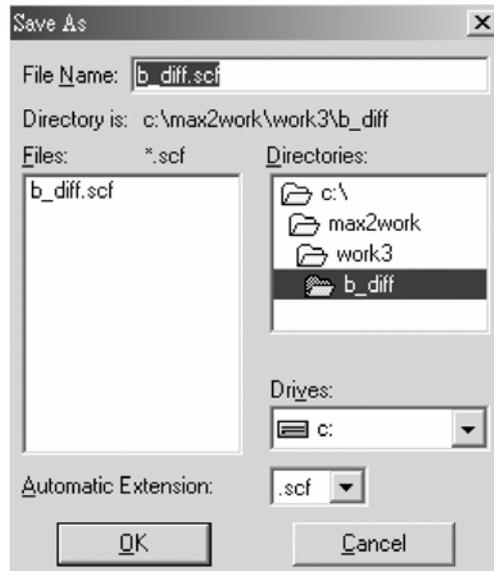


● 圖 4-5-14 模擬結束時間設定視窗

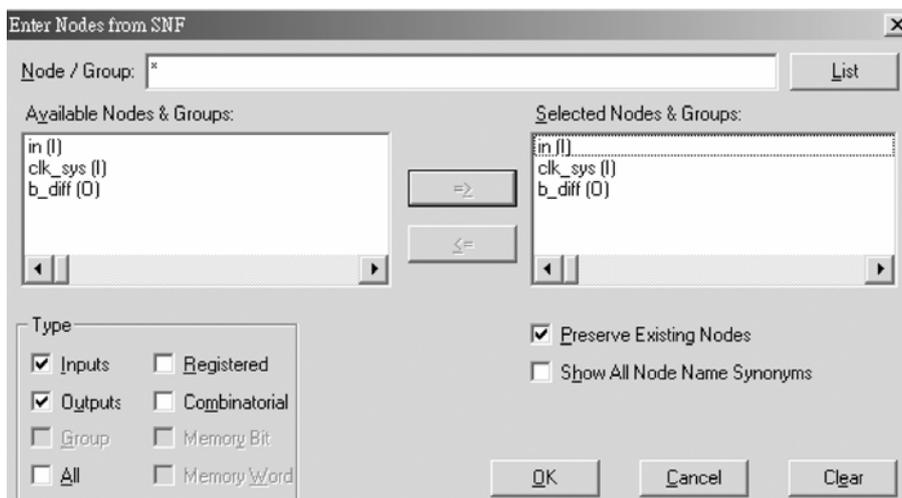


● 圖 4-5-15 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 b\_diff.scf，輸入節點(Node→Enter Nodes from SNF，按 List 及  $\Rightarrow$ ，OK)，編輯輸入信號。

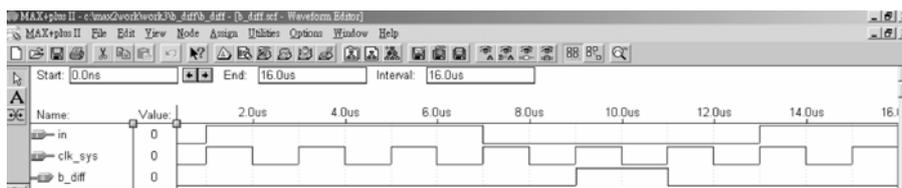


● 圖 4-5-16 儲存檔案視窗



● 圖 4-5-17 輸出入節點選擇視窗

6. 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-5-18 所示，模擬沒有錯誤及警告，所得波形模擬結果符合後微分電路，代表我們製作的電路是正確可用的。



● 圖 4-5-18 模擬結果視窗

### 4-5-3 前後微分電路

如圖 4-5-19 所示為前後微分電路之輸出入波形圖，對照圖 4-5-20 之電路圖，其原理為前微分電路及後微分電路的整合，當輸入信號由 1 變 0 或由 0 變 1 之時，第一個 D 型正反器輸出 0(或 1)，第二個正反器仍保留上一個狀態 1(或 0) 輸出之時，輸出端 fb\_diff 會輸出 1，而且只維持 1 個時脈週期，當下一個時脈週期時，兩個正反器輸出皆為 0(或 1)，此時 fb\_diff 又回復之狀態 0，在此可藉由時脈週期(CLOCK)來調整輸出 fb\_diff 之高態時間長短。

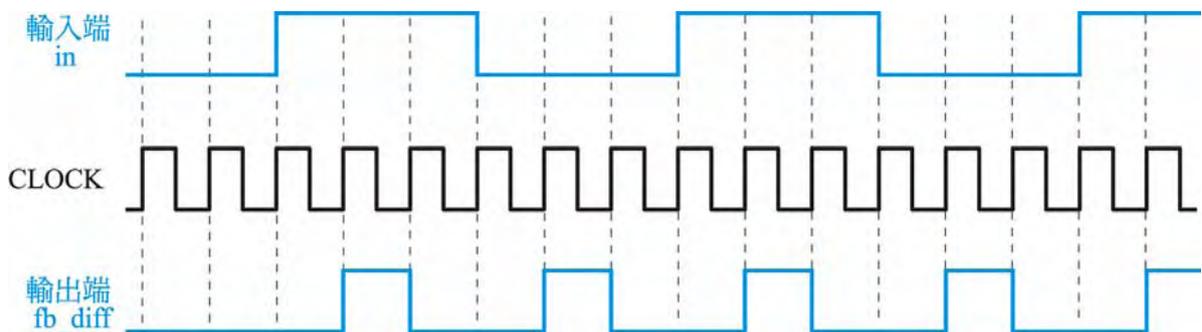


圖 4-5-19 前後微分電路波形示意圖

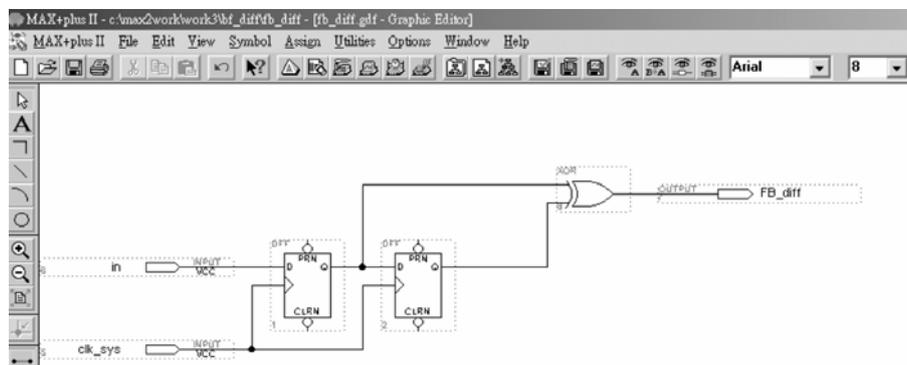


圖 4-5-20 前後微分電路圖

製作前後微分電路步驟如下：

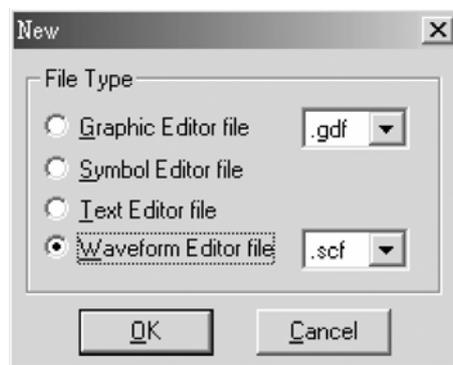
1. 開啟新圖形編輯檔，繪製邏輯電路圖。
2. 新建 fb\_diff6 資料夾後，存檔，取檔名為 fb\_diff.gdf。
3. 設為指定工作專案(File→Project→Set Project to Current File)，指定 CPLD 晶片(Assign→Device)，並編譯(MAX+PLUS II → Compiler)。

如圖 4-5-21 所示，編譯後沒有錯誤與警告，表示邏輯正確，可進行軟體模擬。



● 圖 4-5-21 編譯訊息視窗

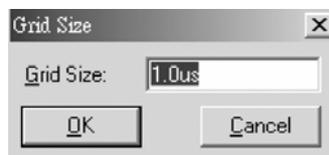
4. 開啟新的波形編輯檔案，設定功能模擬結束時間(File→End Time)設定  $16\ \mu\text{s}$ ，設定格線間距(Options→Grid Size)設定  $1\ \mu\text{s}$ ，顯示在視窗中適當大小格線(View→Fit in Window)。



● 圖 4-5-22 開啟新檔視窗

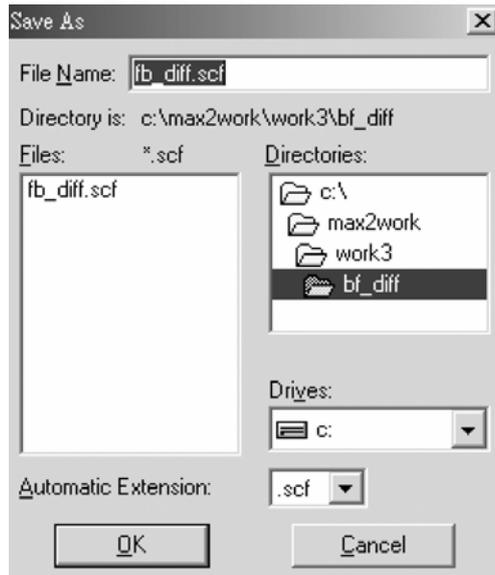


● 圖 4-5-23 模擬結束時間設定視窗

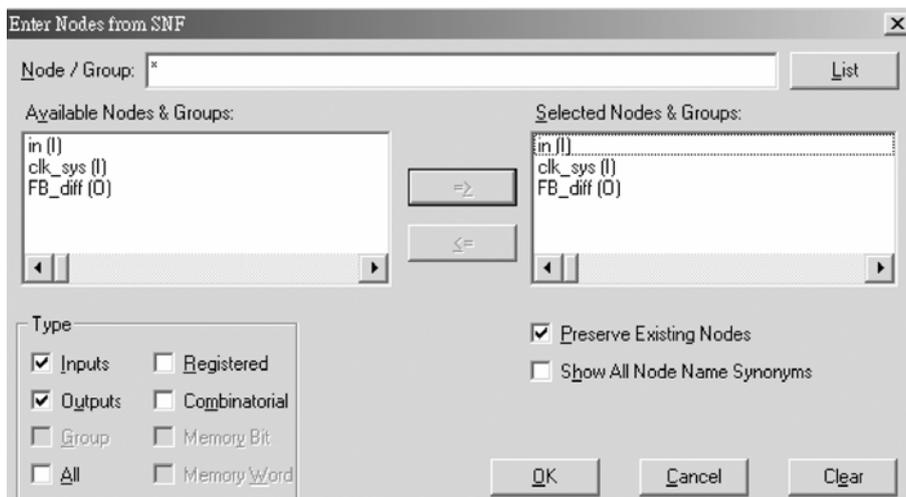


● 圖 4-5-24 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 fb\_diff.scf，輸入節點(Node→Enter Nodes from SNF，按 List 及  $\Rightarrow$ ，OK)，編輯輸入信號。

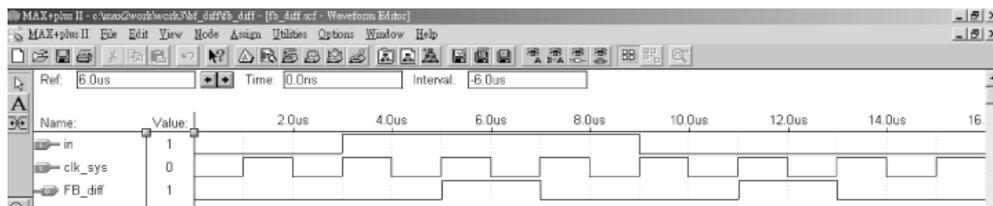


● 圖 4-5-25 儲存檔案視窗



● 圖 4-5-26 輸出入節點選擇視窗

- 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-5-27 所示，模擬沒有錯誤及警告，所得波形模擬結果符合後微分電路，代表我們製作的電路是正確可用的。



● 圖 4-5-27 模擬結果視窗

## 4-6 防彈跳電路

設計電路時，若是有使用到機械式的開關，就不能不考慮的一個重要問題就是開關的彈跳問題。機械式的開關每當開或關之時，由於內部彈簧的慣性，會導致在該瞬間產生多次彈跳，進而影響電路的正確性，因此，防彈跳電路常是電路穩不穩定的一個重要關鍵。

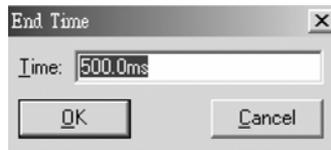
本防彈跳電路利用正反器與邏輯閘組合而成，設計的觀念在於一般開關的彈跳現象大概只會維持在 30 毫秒以下，所以利用二個 D 型正反器檢查此開關的現在狀態及 30 毫秒前的那個狀態，若同時為 0(或 1)，則可假設其開關穩定在 0(或 1)；若是有一為 1，有一為 0 則表示開關正處於彈跳狀態，我們希望開關穩定在 0 時能輸出 0，穩定在 1 時能輸出 1，而處於彈跳時則輸出不能跟著變，而要保持原來狀態，其真值表如表 4-6-1 所示， $SW_n$  表示第一個 D 型正反器的輸出， $SW_{n+1}$  表示第二個 D 型正反器的輸出且其值為第一個 D 型正反器延遲 30ms 之輸出 (因為兩個 D 型正反器接成移位暫存器，時脈信號週期為 30ms)。當  $SW_n SW_{n+1} = 00$  時，我們希望輸出  $Q_{n+1}$  為 0，表示狀態是未按鍵且穩定；當  $SW_n SW_{n+1} = 01$  時，我們希望輸出  $Q_{n+1}$  為維持原來狀態而不改變，即因現在狀態是彈跳狀態，我們不希望輸出跟著彈跳；當  $SW_n SW_{n+1} = 10$  時，我們希望輸出  $Q_{n+1}$  為維持原來狀態而不改變，即因現在狀態彈跳狀態，我們不希望輸出跟著彈跳。當  $SW_n SW_{n+1} = 11$  時，我們希望輸出  $Q_{n+1}$  為 1，表示狀態是未按鍵且穩定；

表 4-6-1 防彈跳電路真值表

$SW_n$	$SW_{n+1}$	$Q_{n+1}$	備註
0	0	0	未按鍵且穩定
0	1	$Q_n$	彈跳狀態
1	0	$Q_n$	彈跳狀態
1	1	1	按鍵且穩定

30ms 取樣一次



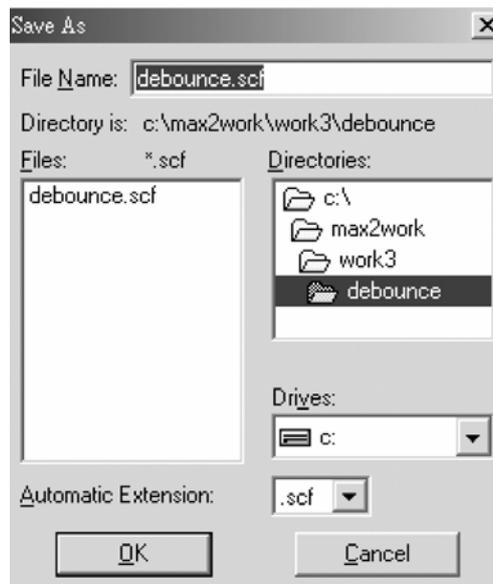


● 圖 4-6-4 模擬結束時間設定視窗



● 圖 4-6-5 模擬單位時間設定視窗

5. 儲存檔案(Save As)，檔名 `debounce.scf`，輸入節點(Node→Enter Nodes from SNF，按 List 及  $\Rightarrow$ ，OK)，編輯輸入信號。



● 圖 4-6-6 儲存檔案視窗

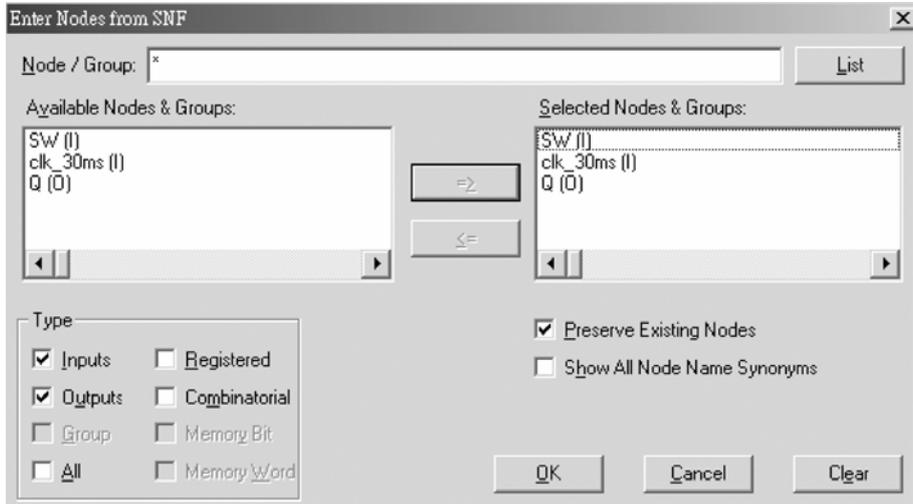


圖 4-6-7 輸出入節點選擇視窗

- 執行模擬(MAX+PLUS II → Simulator, Start)如圖 4-6-8 所示，模擬沒有錯誤及警告，所得波形模擬結果符合防彈跳電路，代表我們製作的電路是正確可用的。

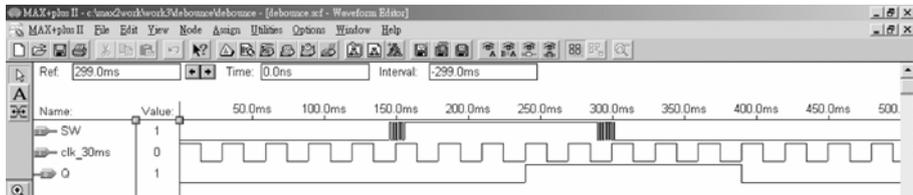


圖 4-6-8 模擬結果視窗

### 4-7 動動腦 順序邏輯練習

1. 請依自行製作元件方法製作前微分電路之元件。
2. 請依自行製作元件方法製作後微分電路之元件。
3. 請依自行製作元件方法製作前後微分電路之元件。
4. 請依自行製作元件方法製作防彈跳電路之元件。
5. 請設計能每按一次鍵上數 1 之四位元上數計數器，並由七段顯示器表示出來。

